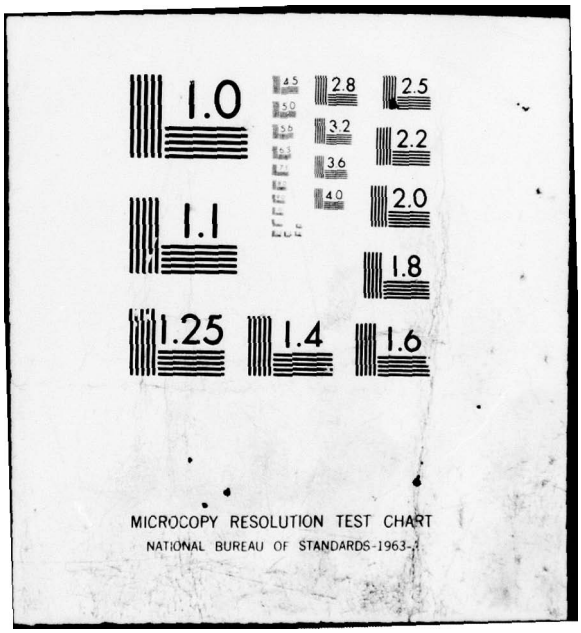
AD-A073 892 MICHIGAN UNIV ANN ARBOR F/G 4/1
A SIMULATION OF THE MIDLATITUDE PLASMASPHERE AND IONOSPHERE.(U)
MAY 79 E R YOUNG, D G TORR, P RICHARDS F19628-77-C-0007
UNCLASSIFIED AFGL-TR-79-0125 NL

1 OF 2
AD
A073892


1.0
1.1
1.25
1.4
1.6
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.8
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AFGL-TR-79-0125

A SIMULATION OF THE MIDLATITUDE PLASMASPHERE AND IONOSPHERE

E. R. Young
D. G. Torr
P. Richards

The University of Michigan
Ann Arbor
Michigan 48109

Final Report
1 December 1976 - 31 March 1979

31 May 1979

Approved for public release; distribution unlimited.

AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: AFGL-TR-79-0125	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): A SIMULATION OF THE MIDLATITUDE PLASMASPHERE AND IONOSPHERE	5. TYPE OF REPORT & PERIOD COVERED: Final Report 1 Dec 1976 — 31 Mar 1979
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): E. R. Young, D. G. Torr, P. Richards	8. CONTRACT OR GRANT NUMBER(s): F19628-77-C-0007
9. PERFORMING ORGANIZATION NAME AND ADDRESS: The University of Michigan, Ann Arbor, Michigan 48109	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61102F, 231162AF, 17, 62
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Geophysics Laboratory, Hanscom AFB, Massachusetts 01731, Monitor/P. J. L. Wildman/PHR	12. REPORT DATE: 31 May 1979
	13. NUMBER OF PAGES: 138
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distrbution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Plasmasphere-ionosphere coupling
Protonosphere
Flux tube
Midlatitude plasma flow and heating

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Plasma and heat flow are simulated for an entire field tube from 120 km in the N. hemisphere spanning the plasmasphere to 120 km in the S. hemisphere.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

228 600

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

1.0 SUMMARY

The summary elements of this report consist of a digest, which follows in section 1.1 and the following elements which appear in appendices: an outline of contract objectives, a cummulative chronological narrative of progress during the duration of the contract, a list of personnel who worked on the contract, a list of scientific talks and a list of publications.

1.1 DIGEST

In this paper we present a simulation of plasma transport of e, H^+ and O^+ along a flux tube of the earth's magnetic dipole field, shown in Figure 1. Heat and particle flux, and temperature and density are simulated, based on realistic geophysical parameters at all times, and at each point of the flux tube.

The model includes a solution of the primary electron and ion production rates, the 2 stream photoelectron equations, ion and electron energy and oxygen and hydrogen ion density equations.

Figure 2 displays the interrelationships of the various components of the model. Control parameters (day, time, longitude, etc.) are used to set up the solution grid and determine gravity, magnetic field strength and various other parameters at each grid point. The neutral atmospheric densities and temperature are then determined. This is followed by the calculation of the primary ionization and photoelectron production rates. From the primary photoelectron distribution the 2 stream equations provide the final photoelectron distribution including transport, and the thermal electron heating rate which is needed in the solution of the energy equations to provide the electron and ion temperatures. Finally, the ion density equations are solved to provide O^+ and H^+ densities at each grid point to complete the procedure for the time step. At the next time step, the procedure is repeated beginning with the neutral atmosphere.

The equations are solved on a spacially varying grid; 300 grid points being used for the L = 3 field line.

The Newton iterative technique has proved to be a stable and economic means of solving the non-linear parabolic partial differential equations of density and temperature. The method is stable enough that time steps can be chosen to fit atmospheric conditions, i.e., stability requirements are not the limiting factor.

Solution of the energy equations on a non-uniform spacial grid has been accomplished by a coordinate transformation based on orthogonal dipole coordinates. The equations are then converted to finite difference form and solved for the ion and electron temperatures using the Newton method. The solution is straightforward and no new numerical problems were experienced.

On the other hand, the solution of the density equations proved to be more difficult. As is widely known, the large diffusion coefficient at high altitudes causes numerical instability as small changes in density generate large changes in velocities. Past methods of overcoming this problem involve setting artificial upper boundary conditions, dividing the field line into regions and using different techniques in different regions, or linearizing the equations.

We have developed a method that solves the coupled O^+ and H^+ equations in a consistent manner for the whole field line. Our method of solution allows us to step over the high altitude region using one complete step whilst maintaining the same basic pattern of solution. By avoiding the high altitude region we avoid the instability. The method is fully explained in a paper which has been submitted to the Journal of Computational Physics. The method is significantly faster than the so called shooting method.

We have further improved the speed of convergence of the Newton procedure by employing prediction techniques. The prediction has the additional benefit of producing a near simultaneous solution of temperature and density even though a ping-pong procedure has been adopted. The prediction provides accurate densities for the next time step which can be used in the temperature solution. A modified steepest descent has been used to guard against the possibility of the Newton procedure overshooting the correct solution and producing negative densities in the process.

The most recent formulation of the transport equations has been used. The code therefore constitutes the most comprehensive tool for modelling the plasmasphere reported to date.

2.0 SCIENTIFIC RESULTS

2.1 Introduction

Study of the thermal plasma population in the closed field line region of the magnetosphere has grown in importance in recent years (see review by Chappell, 1975). Not only is there a significant interchange of plasma between the ionosphere and plasmasphere, but the thermal plasma has also been found to affect the energetic particle population of the magnetosphere. This tight coupling between the magnetosphere-ionosphere system via hot and cold plasma has an impact on energetic particles of the radiation belts, ring current particles and the plasma sheet. Although the broad features have been established (Carpenter and Smith, 1964) a realistic comprehensive model of the plasmasphere, however, has not yet been developed.

It seems clear that the interchange flow processes are related to the shaping of the plasmapause and light ion trough features (Chappell, 1975). The global morphology of the plasmasphere varies strongly with magnetic activity and associated electric fields. When the convection electric field increases during magnetic storms the plasmasphere decreases in size owing to compression on the night side and a peeling off and convective loss of plasma in the afternoon dusk sector. During magnetically quiet periods, the plasmasphere refills as a result of upward flow of cold plasma from the ionosphere, increasing in size. One of the primary objectives of studies of ionospheric-magnetosphere coupling is to relate global morphological storm and quiet time patterns of the behavior of the plasmasphere to those of the ionosphere.

A fundamental requirement for understanding the processes governing the ionosphere-plasmasphere system is a knowledge of the concentration, temperature and flow velocities of charged species populating the field tubes. Since no experiment has yet been devised which can supply these parameters along the full extent of the field tube, we must of necessity resort to models to unify and interpret available and forthcoming fragments of data. Studies of field tube refilling constitute an important approach to the overall problem. Although many studies of this type have been carried out in recent years (these are discussed at greater length below) the details of the filling process still remain unknown, primarily because it has not yet been possible to construct a numerical model which incorporates all salient physical processes.

The purpose of the computer code described in this report is to provide the theoretical framework for quantitative interpretation of measurements of concentration, flow and temperature of thermal ions in the plasmasphere from ongoing and upcoming programs such as Atmosphere Explorer, International Sun-Earth Explorer, Dynamics Explorer, GEOS and several Air Force satellites. The nucleus of the model comprises the solution of the momentum, continuity, energy and heat flow equations for a system of 3 charged species, O^+, H^+ and electrons, i.e., we obtain concentrations, temperatures and flow velocities. Work is currently in progress to include higher order moments for the pressure and stress tensors (Schunk, 1975) to take account of perpendicular and parallel temperatures. The effects of the magnetic gradient force is also being investigated. The equations are solved along an entire field tube with feet located in the E region of the northern and southern hemispheres. Ion production rates and photoelectron heating rates are computed quasi-simultaneously along the field line. Drifts due to neutral winds or electric fields are entered as input data together with other geophysical parameters. As such, the code provides the most comprehensive simulation to date of field aligned plasma transport in the plasmasphere. Nevertheless, at best the code will only provide a basis for

interpretation of observations. It is unlikely that accurate agreement with all data will be attained at this stage, since a greater number of factors will significantly affect the behavior of the plasmasphere than have been incorporated into the current model. Disagreement between model values and measurements will provide a basis for discovering new processes. There are clear indications that significant unidentified plasmaspheric heat sources must exist. Satellite data (Mahajan and Brace, 1969; Sanatani and Hanson, 1970; Serbu and Maier, 1970; Brace et al., 1973; Maher and Tinsley, 1977) indicate large temperature gradients between the equator and ionosphere on a given field tube, for example, which cannot be explained by the photoelectron heat source. This in turn implies that thermal conductivity coefficients must vary significantly along the field tube, otherwise untenably large values of T_e would occur at 1000 km. Wave particle interactions seem to be an obvious candidate. However the problem with wave-particle interactions is that accurate expressions for the "effective collision cross sections", which are needed to correctly describe transport processes such as thermal diffusion and thermoelectric transport, are not available.

The effects of EXB drifts have not yet been satisfactorily treated. When the effects of the latter are considered the magnetic field tube should be allowed to move latitudinally taking account of the volume change in the tube and consequent redistribution of ionization. Ultimately we will require the capability to model the net loss of plasma from the plasmasphere, i.e., in addition to loss to the ionosphere.

We expect that the present model will go a long way toward quantifying the effects of the processes outlined above, and identifying new processes, just as the earlier models of Mayr et al. (1972); Moffett and Murphy, 1973; Massa et al. (1974); Murphy et al. (1976); Bailey et al. (1978) have identified the broad features of tube filling, and the effects of interhemispheric plasma flow on conjugate ionospheres as a function of local time, season and solar cycle.

Of all the hydrodynamic models reported in the literature to date for $L < 1.5$ that of Mayr et al. (1972) was the first to include the momentum, continuity and energy equations in a comprehensive form. It is indeed regrettable that this work was not carried further, as their code inherently contained most of the basic features incorporated in the present work. We have simply improved the formulation of the transport equations and developed an efficient numerical technique for solving the coupled set of equations. However, without the innovations incorporated in the present method, the task is formidable and that may be the reason why the earlier attempts of Mayr et al. (1972) were abandoned.

2.2 Previous Studies

Early work on the general subject of ionosphere-magnetosphere coupling can be broken down into four main categories.

1) Interhemispheric Plasma Exchange: i.e., studies on the interhemispheric flow of plasma and its effects on the plasmasphere and ionosphere.
2) Ionosphere-Protonosphere Coupling: i.e., involving primarily studies of protons escaping from and returning to the ionosphere using a single hemisphere model.
3) General F region and topside ionospheric studies.
4) Formulation of transport equations relevant to the above mentioned studies.

We review progress in these 4 areas below commencing with points (2) and (3) since historically studies of the plasmasphere originated in studies of the ionosphere.

Normal F region

We briefly review work in this area because not only does the behavior of the F_2 layer affect plasma exchange, but the work led to the formulation of the modern aeronomy transport equations and to studies of ionospheric dynamics (discussed in the next subsection) which significantly affect the filling of the plasmasphere after substorm depletion.

Evans (1975) has reviewed recent work on F region dynamics. While the bottom side of the F_1 layer is chemically controlled, diffusion of ionization becomes increasingly important above about 200 km. In the absence of other transport effects hydrostatic equilibrium is established by about 400 km. The plasma diffusion velocity depends upon the ion neutral collision frequency and temperature and pressure gradients of all the ions in the plasma. Early static models (Johnson, 1960; Mange, 1960; Kockarts and Nicolet, 1963; Bauer, 1966 and others) were extended by Walker (1967), and Schunk and Walker (1969) to include thermal diffusion effects. Schunk and Walker presented diffusive equilibrium density profiles calculated for a mixture comprising two major ions, electrons and a number of minor ions. Thermal diffusion was shown to have an important influence on charged particle distributions at altitudes above the F region peak when electron and ion temperature gradients are greater than 1°K/km. For ion and electron temperatures which increase with altitude thermal diffusion acts to drive heavy ions towards higher altitudes, i.e., towards hotter regions. Thermal diffusion also enhances the ordinary diffusion coefficient. Schunk and Walker (1970a,b) extended the work to the diffusion of minor ions in the presence of major ion fluxes, and Schunk and Walker (1973) developed a diurnal model of the E and F regions incorporating these processes.

Models with Neutral Winds

Kohl et al. (1968), Rishbeth (1967, 1968), Stubbe (1968, 1970), Bailey et al. (1969), Abur Robb (1969), Sterling et al. (1969), Strobel and McElroy (1970), Jones (1974) and Roble (1975) and others have developed models which include the effects of winds on the F_2 layer. The winds serve to drive the F region ionization along the field lines. As such the process is important for ionosphere-plasmasphere coupling. The ionization is blown downwards during the day and upwards at night. The subject has been reviewed by Rishbeth (1972, 1974) and Evans (1975).

Winds not only affect the ionosphere, but are affected in turn by ion drag caused by ionospheric plasma which is bound to the field lines at F region altitudes. Therefore several of the above mentioned papers have simultaneously solved the coupled ion and neutral air equations. Stubbe (1970), for example, solved the time-dependent coupled continuity and momentum equations, and electron and ion energy equations for a mixture comprising four ions (NO^+, O_2^+, O^+ and H^+). The photoelectron heating rate was estimated from the ionization rate by using a heating efficiency.

The most comprehensive model developed to date is that of Roble (1975). In Roble's model the primary and secondary photoelectron fluxes, ion production rates and heating rates are computed from the EUV flux and a model atmosphere. Five ion continuity and momentum equations are solved including a flux from the magnetosphere, neutral winds and electric fields. The electron and ion energy equations are also solved using heating rates determined from chemical reactions, photoelectron collisions, and a heat flux from the magnetosphere. The component of the neutral wind along the geomagnetic field is determined from a separate dynamic model of the neutral atmosphere using incoherent scatter radar measurements. Boundary conditions were determined from the incoherent scatter radar measurements of T_e, T_i and O^+ flux at 800 km over Millstone Hill.

Electric Fields

Dynamo electric fields at low to midlatitudes during magnetically quiet periods are less important than neutral winds in controlling the behavior of the F region (c.f., Evans, 1975; Behnke and Kohl, 1974). However, fields of magnetospheric origin during substorms can be significant. Park and Banks (1974) modelled the effect of substorm electric fields which penetrate the plasmasphere. Observations show that a decrease in the layer height at night is often accompanied by an initial increase in $N_{max}F_2$. Park and Banks (1974) show that a downward flow of plasma from the magnetosphere into the F region due to convectional compression of field tubes basically accounts for the phenomenon. Thus electric field effects must be incorporated in any realistic study of field tube filling.

Ionosphere-Protonosphere Coupling

Protonosphere ionosphere coupling has been studied mainly from two complementary viewpoints, namely

1) as a source of ionization in the nocturnal F region.
2) as a source and sink of plasma for the protonosphere.

Several obvious discrepancies in the expected signature of the ionosphere on the plasmasphere lead to studies of proton filling rates in the plasmasphere due to outflow from the ionosphere (Banks et al., 1971). For example, the plasma density decreases on the equatorward side of the light ion trough at L values of 2, whereas the plasmapause lies between L = 4 to 6. From the protonospheric point of view proton escape serves to replenish field tubes that have lost their ionization. The protonosphere from the ionospheric viewpoint therefore is a reservoir of ionization which is filled during the day, and which can act as a source of ionization for the F region at night.

Early theoretical studies predicted significant fluxes of O^+ ions into the F region via the charge exchange process of H^+ with O (Hanson and Patterson, 1964; Geisler and Bowhill, 1965; Geisler, 1967). These predictions were soon confirmed by experimental results (Vasseur and Waldteufel, 1968; Evans, 1969, 1971a,b; Evans et al., 1970; Behnke, 1970; Hagen, 1972). Subsequent theoretical work took account of the relative flow between interacting species and confirmed the importance of the protonosphere as a source of ionospheric plasma (Banks et al., 1971; Schunk and Walker, 1972; Nagy and Banks, 1972; Schunk and Walker, 1973; Moffett and Murphy, 1973; Massa et al., 1974; Murphy et al., 1976). These studies have shed considerable light on filling processes. However, the results are all based on single hemisphere models. In some cases boundary altitudes lay between one to a few thousand km.

The work of Park and Banks (1974) and Murphy et al. (1976) produced several major new results. The former studied the effects of plasma flow into the ionosphere under the influence of neutral winds and an east-west electric field. The latter studied field tube filling after substorm depletion and the subsequent effects on the plasmasphere and ionosphere. Park and Banks (1974) divided the tube extending from the ionosphere to the equator into three regions with specialized formulations for each region. Each region was then coupled to the adjacent region by flux or density boundary conditions. As a result the effect of the total tube content on the ionosphere could be studied under different circumstances. It was found that under steady state conditions plasma flow from the plasmasphere into the ionosphere has a strong stabilizing effect upon the F_2 layer peak density, such that for wide ranges of applied east-west electric fields or north-south thermospheric winds there is essentially no change in N_mF_2. It was also found that N_mF_2 depends sensitively on the plasma density of the plasmasphere. In the time dependent case N_mF_2 may increase significantly due to field tube convection altering the volume of the tube and

squeezing plasma down into the ionosphere. For a midlatitude tube the time needed to achieve a steady state N_mF_2 is much less than the time needed to reduce the tube content appreciably. With a neutral wind such convection does not occur in the time dependent case and N_mF_2 remains effectively unchanged. Park and Banks (1974) point out that this stability of the F_2 layer cannot be achieved without coupled models of the F region and plasmasphere. Therefore all models which rely upon diffusive equilibrium for O^+ in the topside ionosphere will be in error. Since N_mF_2 depends strongly on tube content and since magnetospheric substorms continually agitate the plasma density in the midlatitude plasmasphere, a wide range in N_mF_2 can result.

Murphy et al. (1976) studied the effects of post substorm filling at equinox and solstice for sunspot minimum conditions. They solved the continuity and momentum equations for O^+ and H^+ from 160 to 1400 km and 160 km to the equator respectively. Diffusive equilibrium was assumed for O^+ above 1400 km.

Their results showed a steady build up in tube content for several days after the substorm, and suggested that equilibrium conditions may seldom be realized because of the frequency of occurrence of substorms. Daytime values of the H^+ flux were not affected by the tube content. Nighttime values were found to depend on layer height (N_mh_2) and N_mF_2. The O^+ flux was always large for larger tube content. Tube content continues to increase until the downward H^+ flux balances the upward flux over a diurnal cycle. Prior to this N_mF_2 increases on a night to night basis. The downward H^+ flux depends primarily on

1) Plasma temperature
2) total plasma content of the tube
3) Phase of the neutral wind.

Nighttime downward fluxes were found to have the same stabilizing effect reported by Park and Banks (1974). The effect of upward daytime fluxes is about a 15% reduction of the F region peak density.

Bailey et al. (1978) repeated the calculations for sunspot maximum conditions. They found that H^+ is supplied to the protonosphere during both day and night and results in a net loss of neutral atomic hydrogen. The daily average flux is $\sim 7.5 \times 10^7 cm^2 s^{-1}$.

A special study of interest reported by Bailey et al. (1977) using the one hemisphere model was the theoretical confirmation of counterstreaming of O^+ and H^+ ions observed by Vickery et al. (1976) at twilight from Arecibo. This topic is discussed further for a more general case in the section reporting results obtained with the current model.

Murphy et al. (1976) also assessed the sensitivity of the results to temperature changes. An increase in the gradient of T_e along the field line shortens the filling time. In each of the analysis discussed earlier the electron and ion temperatures were obtained semi-empirically, rather than calculated on the basis of electron densities, etc., as part of the model. The models were essentially exploratory in nature employing many simplifying assumptions and as such served the useful purpose of identifying the main features of the coupled ionosphere-plasmasphere system. The next level of development was the inclusion of interhemispheric flow into the models, and the addition of the electron and ion energy equations.

Interhemispheric Flow of Plasma

Relatively few theoretical studies have treated the interhemispheric flow of plasma, despite the fact that the phenomenon may be important. Some obvious questions that required answers included:

1) Can the direct flow of plasma from one hemisphere to the other affect the conjugate protonosphere and ionosphere. For example, can interhemispheric flow enhance the winter anomaly?

2) Can the redistribution of plasma change upward and downward flow rates?

3) Is there a net transfer of plasma when one hemisphere is illuminated and the other in darkness?

Rothwell (1963), Hanson (1964) and Kohl (1966) discussed interhemispheric flow in connection with the F_2 layer winter anomaly. Nagy et al. (1968) studied interhemispheric flow as a source of heat for the local topside ionosphere. Recent models which included interhemispheric flow have been developed by Mayr (1972), Kutimskaya et al. (1973), Bailey et al. (1978). Murphy and Moffett (1978) use a single hemisphere model but assess the effects of interhemispheric flow. The following main results emerge from these studies.

Mayr et al. (1972) solved the steady state continuity and momentum equations and energy equations for O^+, H^+, He^+ and electrons. The main emphasis of their work was on F region dynamical effects. For example, they found that a wind field assymetric with respect to the equatorial plane, gives rise to assymetric density profiles. Interhemispheric flow decreased the protonospheric density by a factor of 2 in one hemisphere raising the O^+-H^+ transition height by 90 km while increasing the proton density in the other hemisphere. The O^+ and H^+ scale heights changed significantly as did the H^+ flux. A comprehensive study of the latitudinal electron temperature variation showed that model results at 1000 km were in agreement with the measurements of Brace et al. (1967).

A time dependent calculation was also attempted by a perturbation method. The results indicated that diurnal plasma exchange between the protonosphere and F_2 region is significant. As mentioned earlier, these workers did not fully exploit the potential of their model.

Bailey et al. (1978) extended their earlier model to two hemispheres. Despite the fact that the energy equations were not solved, several interesting new results were obtained for sunspot minimum conditions.

1) For midlatitude flux tubes (L = 3) the interhemispheric flow does not directly affect $N_m F_2$.

2) The total content of the plasmasphere is the factor of primary importance. The protonosphere acts as a reservoir which is filled by flows from each hemisphere. Thus, although conjugate ionospheres are decoupled via this reservoir, the longer term filling process provides an important indirect coupling mechanism.

3) Generally there is a significant interhemispheric flow of plasma. Sometimes the flow is from the winter to the summer hemisphere.

4) Although the winter ionosphere is not affected by the direct interhemispheric flow, in general the plasmasphere content tends to enhance winter and reduce summer densities.

5) The flow is upward in both hemispheres during the day.

6) The flow is downward in the winter hemisphere in the evening, but becomes small or changes direction between midnight and dawn.

7) When the flux at 1000 km in one hemisphere is significantly larger than that in the other hemisphere, there is a signature on the interhemispheric flux.

Murphy and Moffett (1978) recently extended their code to include both the energy equations, and to include nonlinear acceleration terms in the momentum equations. Despite comments to the contrary by Bailey et al. (1978) it was found that the solution of the temperature equations yielded unexpected results regarding the effects of gradients on flows. The effect of changes in pressure gradients was approximately cancelled by changes in temperature gradients. Thus the H^+ downflow in a collapsing protonosphere at sunset is not enhanced to the extent expected. Enhancements that do occur are due almost entirely to the decreasing O^+ concentration. Contrary to the suggestions of Fontheim and Banks (1972) no evidence was obtained to suggest that supersonic flows occur.

Formulations and Solution of the Transport Equations

The basic equations which describe the temporal and spatial behavior of rarefied gases are expressed in terms of their concentrations, bulk flow velocity and temperature. These quantities are included within the basic conservation equations derived by taking velocity moments of various terms of Boltzmann's equation. Schunk (1975, 1977) points out that the basic equations used in modelling the ionosphere and protonosphere have not, in general, been applied with a clear recognition of their intrinsic limitations. Attempts have been made to study ionospheric-magnetospheric plasma exchange using the binary formulations of Chapman and Cowling (1960) to obtain drift velocities for the multicomponent O, O^+, H^+ gas mixture (Massa, 1974; Murphy et al., 1976). Schunk and Walker (1970a) have investigated the accuracy of the binary formulation and have shown that while it is quite accurate for the derivation of the ambipolar diffusion coefficient, diffusion coefficients for minor ions can be in error by as much as a factor of 2. The factor of 2 arises from the fact that the binary approximation is less accurate for Coulomb collisions than for ion-neutral collisions. Since the major ions in the topside ionosphere go through a transition from O^+ major in the F_2 region to H^+ major in the protonosphere, it is essential not only to describe the minor ion diffusion processes correctly, but furthermore, to have a system of equations which accurately describes the behavior of the gas mixture in the transition region. It is also desirable, in order for the calculation to be accurate, that thermal diffusion effects be included accurately in our transport equations, as was shown by Schunk and Walker (1970b).

Schunk and Walker (1970b) used the multicomponent formulation of Hirschfelder et al. (1964) to derive the continuity equations for a three-constituent mixture comprising O, O^+ and electrons. Although this approach yields diffusion coefficients which are accurate to within 20% (Schunk and Walker, 1970b), as the numbers of gases in the multicomponent mixture increases

it becomes impossible to manipulate the equation without an unreasonable amount of effort, if the 20% accuracy is to be retained.

An alternative approach to transport equations derived by Burgers (1969) has recently been presented by Schunk (1975) for applications to aeronomic studies. This system has many advantages over the systems previously used for the formulation of the plasma exchange problem between the ionosphere and the protonosphere.

Schunk's (1975) system of equations used Grad's (1949, 1958) 13-moment approximation for the distribution function of each of the species that constitute the plasma. The basic advantages of Schunk's formulation stem from the fact that interactions between one species and any other species are described by collision integrals that are evaluated to the same degree of approximation as the distribution function themselves. The approach appears to yield accuracies equivalent to the second order "multicomponent" formulation. However, it has several advantages over the multicomponent formulation used by Schunk and Walker (1970b). There are two important advantages that we see in Schunk's (1975) formulation.

The first important feature although not really a new one, is that the equations can be handled directly numerically without having first to solve for diffusion coefficients and other similar parameters. For example, in Schunk's (1975) 13-moment formulation, transport properties of the gas are described by first order equations, namely continuity, momentum, energy, pressure tensor and heat flow. Thus the heat flow terms that appear in his momentum collision term account for thermal diffusion effects, while the drift velocity terms in the heat flow collision term account for thermoelectric and diffusion-thermal effects. Thus instead of solving in terms of a thermal diffusion coefficient one can solve the coupled system of equations directly.

A second feature of interest that stems from Schunk's (1975) system of equations is that any given constituent is allowed to have its own temperature and drift. This enables us for the first time to study in detail the effects of a temperature gradient in one species on the heat flow, or thermal diffusion of another species with a different temperature gradient.

At present we have not included all the features discussed above in the present code, but have adopted the modified momentum equations of St. Maurice and Schunk (1977) and energy equations of Schunk and Nagy (1978) as appropriate for our current concepts of the quiet plasmasphere (Schunk and Watkins, 1979).

Application of this system to the midlatitude ionosphere and plasmasphere permits several simplifying assumptions to be made which significantly reduce the complexity of the equations.

We consider a plasma comprised of only two major ions, O^+ and H^+ and electrons in a neutral atmosphere of O, O_2, N_2 and H. We assume that species temperature and flow velocity differences are small, a midlatitude condition. Thus we can neglect stress and nonlinear acceleration terms, and use Burger's (1969) linear collision terms (c.f., St. Maurice and Schunk (1977)). In addition density and temperature gradients perpendicular to the geomagnetic field lines are neglected, and we have assumed that ion and electron temperature distributions are isotropic, so that we can ignore the stress tensor equation.

Even though many of the individual terms in our equations arise directly from Schunk's 13-moment system of equations, we have arrived finally at a basic formulation involving only six equations. These are: the photoelectron two stream equation, the electron energy equation, the ion energy equation, St. Maurice and Schunk's velocity solution of the ion and electron momentum equations, and finally the continuity equations. We might add that at low to intermediate altitudes, where it is appropriate, we have included ion neutral collitions with the equations of St. Maurice and Schunk in order to solve for the individual

ion velocities separately. These basic equations of our plasma formulation are given collectively in (1) - (6) in the section on Basic Equations.

The primary photoelectrons are treated via a two stream Liouville equation (Banks and Nagy, 1970) which essentially follows their distribution through phase space until they are thermalized. One hundred energy steps are used at each spatial grid point separately for the upstreaming and downstreaming electron populations. Such a detailed phase space treatment is necessary only for the highly superthermal photoelectrons.

Currently neutral winds can be input numerically, or the code could be modified to include an analytical fit to experimental or model wind values such as those of Blum and Harris (1975). Provision has not yet been made for electric field convection of tubes of flux. Despite significant advances over the last decade, which were described earlier as well as in a number of reviews (Carpenter and Park, 1973; Chappell, 1972; Banks, 1972; Banks et al., 1976), the plasma coupling mechanisms between the ionosphere and magnetosphere have not been studied using a realistic formulation which simultaneously accounts for variations in all the major controlling variables.

Another major shortcoming in previous studies has been the use of ad hoc boundary conditions, which generally arise when only part of a field line is modeled. Roble (1975) and Stubbe (1970) formulated their coupled parabolic equations for O_2^+, NO^+, O^+ and H^+ in the F region and topside ionosphere, but H^+ has been treated as a minor ion in each of these cases, or the H^+ density is calculated by assuming chemical equilibrium with O^+. Such formulations all apply to altitudes below about 2000 km, and require artificially designated or measured upper boundary conditions.

Bauer (1968) and Massa (1974) attempted to extend this treatment for O^+ and H^+ along the entire flux tube but this leads to serious numerical problems, which are discussed in the next section. We show in the next section that a

satisfactory solution to the entire field tube problem can be achieved by linking appropriate low and high altitude formulations with continuity conditions on flux and density at about 1500 km.

The searching method has also been used in attempts to treat the entire flux tube by a single method. In the searching method (Moffett and Murphy, 1973; Mayr et al., 1965; Murphy and Moffett, 1978; Bailey et al., 1977; Richards, 1978) the flux or velocity from the continuity equation is substituted into the collision terms of the momentum equation. The resulting integro-differential equation is then integrated numerically down from the equator where flux and density boundary conditions are imposed. The equatorial boundary conditions are readjusted and the integration performed repeatedly until low altitude chemical boundary conditions are satisfied. The main disadvantages of this approach stem from the fact that flux determinations from the continuity equation are inherently inaccurate and unstable at low altitudes, and that the method does not produce simultaneous solutions in both hemispheres. Therefore, convergency may be up to an order of magnitude slower than that discussed in this paper.

The perturbation solution of Mayr et al. (1972) also treats the entire field line as a single region. Mayr et al. integrate numerically but with only a gradually increasing fraction of the collisional terms in the momentum equations, along the entire field line. The drag terms are always evaluated in the n'th approximation to obtain the n + 1st approximation. This method also suffers from low altitude problems for the same reasons mentioned in connection with searching methods. It is apparent that both of these methods are basically high altitude formulations and that neither should be used unless an accurate lower boundary at, say, 1000 km can be supplied by some other type of calculation or by measurement.

Multiple region formulations somewhat similar to our own have been used by Park and Banks (1974) and Murubashi and Grebowski (1976), who treated the

topside region above 3000 km as a reservoir. The former treatment lacked low altitude diffusion (1974), however, and the latter ignored significant terms in the diffusive equilibrium formulation at high altitudes (1976). We surmount these limitations in our treatment by simulating the flow and density in an entire flux tube spanning the midlatitude plasmasphere between magnetically conjugate points in the F regions of the northern and southern ionospheres, as shown in Figure 1. The solution encompasses several regions of differing dominant physical processes, which in turn require different formulations and numerical treatments. We use an optimal mathematical approach for each altitude regime, and derive the appropriate and unique boundary conditions which link these regions.

In order to produce a geophysically meaningful and consistent simulation, a number of parameters must be computed in the proper sequence as shown in Figure 2 which is an overall schematic of our simulation code. A line with an arrow in the schematic indicates a parameter that is supplied by one program or group of programs to another program or group of programs. Lines without arrowheads indicate continuity equations. Figure 2 shows how the earth's rotating tilted dipole, gravitational and magnetic field are calculated locally in the proper sequence and neutral atmospheric parameters may be taken locally from a semi-empirical model. We selected the MSIS model atmosphere by Hedin *et al.* (1977a,b) because it reproduces the seasonal solar cyclic and diurnal variations of the atmosphere as well as its response to geomagnetic disturbances.

2.3 Basic Equations

Photoelectron Liouville Equation

$$B \frac{d}{ds} \frac{\phi^+}{B} = - T_2 \phi^+ + T_1 \phi^- + \frac{q}{2\langle\cos\theta\rangle} + \frac{q^+}{\langle\cos\theta\rangle}$$

$$-B \frac{d}{ds} \frac{\phi^-}{B} = - T_2 \phi^- + T_1 \phi^+ + \frac{q}{2\langle\cos\rangle} + \frac{q^-}{\langle\cos\theta\rangle}$$

where $\phi^{+}(E,s)$ = photoelectron flux outward along s

$\phi^{-}(E,s)$ = photoelectron flux inward along s

$q(E,s)$ = photoelectron production rate in the range E to E + de due to direct ionization processes.

$q^{\pm}$ = photoelectron production in the range E to E + de due to cascading from higher energy photoelectrons undergoing inelastic collisions.

$\langle\cos\phi\rangle$ = average cosine pitch angle

B = magnetic field strength

$T_1 = \sum_k n_k p_e^k \sigma_e^k \qquad T_2 = \sum_k n_k [\sigma_a^k + p_e^k \sigma_e^k]$

and

n_k = k^{th} species number density

p_e^k = photoelectron backscatter probability for elastic conditions with the k^{th} species.

σ_e^k = photoelectron total scattering cross section for elastic conditions with the k^{th} species.

σ_a^k = inelastic cross section for excitation of the k^{th} particle species.

Thermal Electron Energy

$$\frac{3}{2} N_e k \frac{\partial T_e}{\partial t} = - N_e k T_e \nabla \cdot U_e - \frac{3}{2} N_e k U_e \cdot \nabla T_e - \nabla \cdot q_e + \Sigma Q_e - \Sigma L_e \tag{3}$$

Ion energy

$$\frac{3}{2} N_i k \frac{\partial T_i}{\partial t} = - N_i k T_i \nabla \cdot U_i - \frac{3}{2} N_i k U_i \cdot \nabla T_i - \nabla \cdot q_i + \Sigma Q_i - \Sigma L_i \tag{4}$$

Ion Heat Flow

$$q_1 = \frac{1}{1-\xi} \left(- \lambda_i \nabla T_i - \frac{P_i}{P_j} \frac{\nu_{ij}}{\nu_i} \lambda_j \nabla T_j \right) \tag{5}$$

Electron Heat Flow

$$q_e = - \lambda_e \nabla T_e \tag{6}$$

Ion Electron Momentum

$$\begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} = - \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \frac{\begin{bmatrix} (1 + \frac{\nu_{2x}}{\nu_{21}}) & 1 \\ 1 & (1 + \frac{\nu_{1x}}{\nu_{12}}) \end{bmatrix} \begin{bmatrix} D_1 & Q_1 \\ D_2 & Q_2 \end{bmatrix}}{(\frac{\nu_{1x}}{\nu_{12}} + \frac{\nu_{2x}}{\nu_{21}} + \frac{\nu_{1x}}{\nu_{12}} + \frac{\nu_{2x}}{\nu_{21}})} \tag{7}$$

Ion Continuity

$$\frac{d}{dt} \begin{bmatrix} N_1 \\ N_2 \end{bmatrix} = \begin{bmatrix} P_1 \\ 0 \end{bmatrix} - \hat{L} \cdot \begin{bmatrix} N_1 \\ N_2 \end{bmatrix} - \nabla \cdot \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} \tag{8}$$

where

$N_e/N_1/N_2$ = electron/O^+/H^+ density

$T_e/T_1/T_2$ = temperatures

$U_e/U_1/U_2$ = average drift velocities

q_e/q_i = electron/ion heat flux

$\Sigma Q_e/\Sigma L_e$ = sum of electron heating/cooling rates

and we have used the thermal conductivity coefficients (λ_i, λ_e), the diffusion coefficients (D_i) and terms in the diffusive force, which we symbolize by Q_i, from St. Maurice and Schunk (1977). The term ξ in the denominator of (5) is given by St. Maurice and Schunk as

$$\xi = \nu_{ij}' \nu_{ji}' / (\nu_i' - \nu_j') \tag{9}$$

where the ν_{ji}', ν_{ij}', ν_i' and ν_j' are their "effective collision frequencies" Also we may collect the diffusive force terms given by them in the form

$$Q_i = \frac{\nabla(N_i T_i)}{N_i T_i} - \frac{m_i g_{||}}{kT_i} + Z_i \frac{T_e}{T_i} \frac{\nabla(N_e T_e)}{N_e T_e}$$

$$\mp \gamma_i \frac{\nabla T_e}{T_i} \pm \frac{N_j}{N_e} \left(\frac{\alpha_{ij}}{T_i} \nabla T_i - \frac{\alpha^*_{ij}}{T_i} \nabla T_j \right) \tag{10}$$

where the upper/lower sign applies to O^+/H^+; α_{ij}, α^*_{oj} and α_i are thermal diffusion coefficients and D_i is the ordinary ion diffusion coefficient of species i, as in St. Maurice and Schunk. The matrix L in the continuity equation contains all the chemical production and loss frequencies for both ions.

2.4 Numerical Simulation

Outline of Approach

Equations (3) to (7) form a system of 8 first order differential equations. Equations (5) and (6) can be substituted into (4) and (3) respectively to form two parabolic partial differential equations. Similarly equation (7) can be substituted into (8) to yield 2 parabolic equations. Two boundary conditions required for each parabolic equation and these are supplied at the end of the field tube. The boundary altitude can be chosen to ensure that local equilibrium conditions prevail. This approach has worked successfully for the solution of the energy equations. In the case of the continuity equations, numerical problems are encountered above some altitude which generally lies between 1000 and 3000 km depending on prevalent conditions. In this region the diffusive force term, Q_i, given by (10) tends to zero while the diffusion coefficient, D_i becomes very large although the ion fluxes remain finite. The individual terms of Q_i do not become small at high altitudes. Therefore, Q_i becomes the small algebraic sum of several relatively larger terms. The fractional error of Q_i becomes large at high altitudes, and so does the fractional errors of ion flux, which is a linear combination of Q_i terms when calculated from the momentum equations. The fact that $Q_i \simeq 0$ implies that diffusive equilibrium prevails. We therefore use equation (10) to compute the O^+ and H^+ densities in the region where $Q_i \simeq 0$.

We divide the plasma flux tube into three regions B, C, and B* shown in Figure 1. In the low altitude regions B and B* collisions are important and the full parabolic equations are used. These equations accurately reproduce the plasma concentration and flow in this region. In region C, the plasma may be termed collisionless and the diffusive equilibrium approximation is used. This procedure is effectively equivalent to using the shooting method in region C. The approach eliminates the low altitude problems which are peculiar to the shooting method, as well as the high altitude problems which limit the

parabolic approach. The solutions for each region are coupled through a flux preserving approach which is described in detail in subsequent sections.

Although the density profiles calculated in region C have the shape of diffusive equilibrium profiles at any instant, they match the time variable densities at boundaries 2 and 2* (see Figure 1). Therefore non zero fluxes are required to inflate or deflate the field tube in this region. On the basis of this fact we avoid the use of the term "diffusive equilibrium" and adopt the terminology "dynamic equilibrium" instead.

If we consider regions B and B* as separate regions whose ion densities are described by coupled parabolic second order equations, it is clear that one boundary condition per variable is required at each of the respective upper boundaries 2 and 2*, in order to obtain a unique mathematical solution. If regions B and B* were adjacent it is clear that each would supply the boundary for the other. With separated regions it is equally valid to choose two independent equations per variable, each describing a relation between the values of fluxes and densities at the boundaries 2 and 2*. We have chosen to numerically integrate the equations of dynamic equilibrium (10), over region C to provide one boundary relation per variable. This procedure provides the density at boundary 2*, as well as the total flux tube contents as a function of the density at boundary 2. We then integrate the continuity equations for H^+ and O^+ over region C to obtain the second independent boundary relation for each variable, this time relating the boundary fluxes at points 2 and 2* to the time variation of the total flux tube content.

As the continuity equations and diffusive equilibrium equations are physically valid as well as independent, it is clear that the final solution will be geophysically meaningful as well as mathematically unique.

Flow Chart Description

In Figure 2, the overall schematic of our simulation, each block represents a routine or model providing necessary geophysical parameters or simulating some aspect of plasma transport, i.e., routines were written to supply parameters related to the magnetic field geometry, the neutral atmosphere, ion production rates, heat sources and so on.

As a first initialization step, the geophysical parameters defining the problem such as UT date and time, L shell, magnetic longitude, and solar and magnetic activity indices are input via terminal command, or equivalent batch input. Then the geometry and fields routines calculate an appropriate spatial grid along the specified field line of the earth's rotating tilted dipole, and at each point compute the magnetic and gravitational fields and the solar zenith angle. The grid is constant step size in a modified dipole coordinate system which results in much smaller actual spatial steps at low altitudes where scale heights are smaller. A variable step size flux preserving code is used with the spatial steps to solve the density equations while a standard equal step finite difference code is used with the modified dipole coordinates to solve the temperature equations. Next, the initial profiles of ion density, N_i, ion temperature T_i and electron temperature T_e, are input from stored data and interpolated if necessary to match the grid points for the problem at hand. This is the last initialization step. The steps of the simulation described hereunder must be repeated with each time step.

The first repeated step is to obtain the neutral temperature and densities from the MSIS model at each spatial grid point for the particular time of the step being simulated. The primary photo-electron production in 100 energy intervals is calculated using neutral densities from MSIS, solar fluxes or Hinteregger (1978) and photoionization cross sections listed by Kirby Docken et al. (1979). The degradation of the primary photoelectron spectrum is

then computed using the 2-stream equations of Banks and Nagy (1970). The temperature and density profiles of the thermal ions and electrons are taken from the previous time step except that initial profiles are used for the first time step. The 2-stream equations supply the thermal ion and electron heating rates required by the thermal energy equations.

The thermal energy equations form a coupled parabolic system. Solution of these equations by an iterative technique yields the ion and electron temperatures at the end of each time step. The ion momentum and continuity equations, enclosed within a larger dashed block in Figure 2, are solved simultaneously by an iterative procedure and intereact as a unit with the other elements of the code. Referring to Figure 2 we see that they require the previously calculated ion and electron temperatures and produce, in their turn, the ion density, velocity and flux profiles used by the energy and 2-stream equations of the next time step.

The Minimization Function of the Density Equations

A Newton iteration similar to that described by Hastings and Roble (1977) is used to solve the finite difference equations. An important consideration is the choice of the function F to be minimized. As will become evident from the discussion that follows the most suitable choice for F is an integral form of the coupled parabolic density equations. The integration is carried out between limits located midway between grid points, e.g., if the function at the kth point is denoted by F_k, the integration is carried out between points k-1/2 and k+1/2, i.e.,

$$F_k = \frac{1}{B(S)} \begin{bmatrix} N_1 & 0 \\ 0 & N_2 \end{bmatrix} \left[\frac{\begin{bmatrix} (1 + \frac{\nu_{2x}}{\nu_{21}}) & 1 \\ 1 & (1 + \frac{\nu_{1x}}{\nu_{12}}) \end{bmatrix}}{(\frac{\nu_{1x}}{\nu_{12}} + \frac{\nu_{2x}}{\nu_{21}} + \frac{\nu_{1x}}{\nu_{12}} \frac{\nu_{2x}}{\nu_{21}})} \right] \begin{bmatrix} D_1 & Q_1 \\ D_2 & Q_2 \end{bmatrix} \Bigg|_{S_k - 1/2}^{S_k + 1/2}$$

$$+ \int_{S_k - 1/2}^{S_k + 1/2} \left\{ \begin{bmatrix} P_1 \\ 0 \end{bmatrix} - \hat{L} \cdot \begin{bmatrix} N_1 \\ N_2 \end{bmatrix} - \frac{\partial}{\partial t} \begin{bmatrix} N_1 \\ N_2 \end{bmatrix} \right\} \frac{ds}{B(S)} = 0 \qquad (12)$$

Between grid points in regions B or B* densities are interpolated linearly in evaluating (12). This approach lends itself naturally to the use of integration steps of arbitrary size with automatic conservation of total ion count. The approach may be termed a "flux preserving method". It is the choice of this approach which renders the linking of the second and first order equations a simple procedure. The use of an integral form of the continuity equation therefore simplifies the inclusion of region C in an iterative solution scheme. In this case we integrate from boundary 2 to boundary 2*, across the entire region C.

$$F = - \left\{ \frac{1}{B(2*)} \begin{bmatrix} \phi_1(2*) \\ \phi_2(2*) \end{bmatrix} - \frac{1}{B(2)} \begin{bmatrix} \phi_1(2) \\ \phi_2(2) \end{bmatrix} \right\}$$

$$+ \int_2^{2*} \left\{ \begin{bmatrix} P_1 \\ 0 \end{bmatrix} - L \cdot \begin{bmatrix} N_1(S) \\ N_2^1(S) \end{bmatrix} - \frac{\partial}{\partial t} \begin{bmatrix} N_1(S) \\ N_2^1(S) \end{bmatrix} \right\} \frac{ds}{B(S)} \qquad (13)$$

where the densities at the many grid points between boundaries 2 and 2* are no longer interpolated, but rather calculated from the densities at point 2 via numerical integration of physically realistic equations of diffusive equilibrium. This may be expressed in functional form as

$$N_i(k) = G_i\ (N_1(2),\ N_2(2),\ k) \tag{14}$$

at some point k of region C. The fluxes used in the continuity equation of region C are calculated via the full momentum equations which are still valid at the boundaries 2 and 2*.

The Finite Difference Scheme

We use the following scheme to evaluate the time average of the parameter f, and its spatial derivative $\frac{\partial f}{\partial s}$ at a point midway between grid points k and k+1 during the time interval t_ℓ to $t_{\ell+1}$.

$$f(S_{k+\frac{1}{2}}) = \frac{\theta_f}{2} f(S_k, t_{\ell+1}) + \frac{(1-\theta_f)}{2} f(S_k, t_\ell) + \frac{\theta_f}{2} f(S_{k+1}, t_{\ell+1}) + \frac{(1-\theta_f)}{2} f(S_{k+1}, t_\ell) \tag{15}$$

$$\left.\frac{\partial f}{\partial t}\right|_{k+1/2} = \theta_f \frac{[f(S_{k+1}, t_{\ell+1}) - f(S_k, t_{\ell+1})]}{S_{k+1} - S_k} + (1-\theta_f) \frac{[f(S_{k+1}, t_\ell) - f(S_k, t_\ell)]}{S_{k+1} - S_k} \tag{16}$$

The parameter θ provides the option to select the finite difference scheme, with θ = 1/2 for Crank-Nicholson and θ = 1 for Laasonen differencing for example.

Referring to Figure 3, we see that the computation of eq. (12), totally within one of the regions B or B* is based upon the ion densities at three adjacent grid points, each one of which receives an independent correction in the iterative solution. A polynomial interpolation is used to interpolate the densities between 3 consecutive grid points. The computation of equation (13) on the other hand involves a number of dependent densities at intervening points as well as the densities at points 2B, 2A and 2B*. The boundary fluxes φ(2) and φ(2*) are each calculated from the densities at the adjacent point pairs (2B, 2A) and (2A*, 2B*), respectively. It is to be noticed however that the ion densities at point 2A* are not independent, but computed via the equations of diffusive equilibrium from the densities at point 2A. The integral terms also involve dependent ion densities at all the grid points between 2A and 2B*.

In summary, even though equation (13) involves all the grid points of region C, only the densities at the points 2B, 2A and 2B* receive independent corrections during the iterative solution procedure. In principal the entire region C acts as a single "heavy point", keyed to the density at 2A, in the Newton Raphson solution of the density equations.

The use of an integrated continuity equations makes the adoption of a variable step size practically automatic as mentioned above. Short steps (3-4 km) must be used near the lower boundaries, while steps of several hundred km may be used in the plasmasphere, and the entire region C serves as a single long step in the middle.

The iterative solution to the temperature equations is obtained differently than that for the continuity and momentum equations discussed above. When the ion and electron heat flow equations (5) and (6) are substituted into the ion and electron energy equations (3) and (4), the latter become parabolic equations in the temperature. For these equations we have expressed the spatial step in terms of the modified dipole coordinates which were used to determine the grid points along the field line. In terms of the dipole coordinates, the step size is actually a constant, and the equations of temperature are expressed in the standard finite difference form and then solved by a Newton iteration in which all the grid points of region C receive independent corrections.

2.5 Control Options

In order to adapt the code to the simulation of a variety of geophysical conditions, locations and analysis procedures it is desirable to be able to adjust a number of model parameters and options with each run. The control options of this code fall naturally into the three categories of geophysical activity, geometry, and numerical analysis as outlined in Table 1. The

planetary index A_p, the 10.7 cm solar flux and the daily average 10.7 cm flux serve to define the state of solar and geophysical activity. They are used internally to make appropriate adjustments in the MSIS neutral atmosphere model (see Figure 2).

Referring again to Figure 1, the geometrical definition parameters include the L shell and magnetic longitude, which determine the location of the chosen magnetic flux tube relative to the earth. The U.T. date and time and the ephemeris transit, then precisely determine the rotation of the earth and flux tube about the geographic axis. Finally with the declination, we fully determine the position and orientation of the magnetic flux tube relative to both the earth and the sun.

The first two numerical analysis parameters required are the boundary altitudes Z(1) and Z(2), which define the regions to which we must apply our high and low altitude formulations. The number of grid points and the point distribution parameter suffice to determine the position of individual points along the flux tube. They may be concentrated at low altitudes in a smoothly varying fashion or spread out more evenly as desired. We have been able to simulate an L=2 flux tube with only 250 points. A sequence of variable size time steps must also be defined for the simulation. With time and spatial steps now completely defined the code internally calculates all of the ambient parameters of the environment which are needed at each time and point of the plasma simulation. Referring to Figure 2 we see that all the geometrical and field quantities may be calculated, the neutral atmospheric densities and temperature and the primary photoelectron production, with the input of the above control parameters.

TABLE 1

Control Options and Parameters

Input Parameters

Geophysical Activity

A_p

F 10.7

F 10.7

Output Parameters

Various print options are available:

half or all of the profile print at

every n^{th} grid point

Geometrical Definition Parameters

L Shell

Magnetic Longitude

UT Date

UT Time

Ephemeris transit

Declination

Numerical Analysis Parameters

Z(1) Boundary Altitude

Z(2) Boundary Altitude

Number of spatial grid points

Point distribution parameter

Sequence of time steps

Implicit fraction in finite differencing

Time dependent simulation option

Steady state simulation option

The code may be run with Crank-Nicholson, implicit or any type of intermediate finite differencing scheme under the control of the implicit fraction parameter. Thus if θ_f is input, N_j would be approximated as $\theta_f N_j (t + \Delta t) + (1 - \theta_f) N_j(t)$ and so on. Either fully time dependent or steady state simulation may be selected via another parameter which artificially nulls the explicit time derivative term in the ion continuity equation.

Finally, we have several output options. The basic outputs are the ion densities and velocities and the ion and electron temperatures at the grid points. Print options include printing half or all the profile at selected grid points, e.g., every third point. Optional outputs include individual terms in the ion continuity and momentum equations, which have proved very useful in analyzing the causes of simulated ion flows and densities. These printouts may be for any specified point count range or the entire field line.

2.6 Typical Results

We first discuss the tests we performed to demonstrate that the simulation converges to a stable steady state solution. We then analyze the steady state ion fluxes, and finally we present simulations of the collapse of the sunset ionosphere. This work has been reported by Young *et al.* (1979a).

For the stability test, the rotation of the earth and the time evolution of all locally computed parameters were frozen at UT 17:33 Aug 10, and furthermore set artificially to N-S symmetry about the magnetic equator of an L = 2 field line at approximately 70 W longitude. Our temperature model for this test featured T_i = 4500°K, T_e = 5500°K at the magnetic equator and $T_i = T_e$ = 500°K at 120 km. To obtain a steady state solution the time derivative terms of equations (12) and (13) were set equal to zero and the simulation allowed to iterate to convergence.

Using the steady state solution as an initial condition we then allowed the simulation to run in the time dependent mode, but with all the local ambient parameters still frozen at UT 17:33. Using many 5 minute time steps we found that the time dependent density profile simulation did not drift and displayed an oscillation of less than one part in a million about the steady state solution. The profiles also proved to be symmetric to better than one part in a million. We feel that the results of these tests constitute a rigorous test of stability and conservation of particles (Young et al., 1979a).

The steady state profiles show some interesting aspects of ion production and flow which indicate that the code is reproducing expected geophysical variations. The steady state flux profiles, denoted 0h in Figure 4, show downflow of both O^+ and H^+ below 500 km but counterstreaming above with O^+ moving upwards and H^+ downwards. The peak O^+ downflux occurs very strongly near the altitude of the O^+ density peak at about 250 km as shown in Figure 5. This can be attributed to the fact that the chemical loss rate, due to reaction of O^+ with N_2 and O_2 increases with decreasing altitude more rapidly than the photoionization production rate as one approaches the O^+ density peak from above. The net result is that the O^+ ions are actually flowing into their region of most rapid production (see Young et al., 1979a).

Above about 500 km, the calculated O^+ flux is upward, as is typically seen in daytime measurements (Evans and Holt, 1978) and the H^+ is counterstreaming downward. At altitudes above 550 km, the counterstreaming H^+ and O^+ flux are virtually equal in magnitude except for a small residual difference caused by the small amount of photoproduction even at very high altitudes.

Photoproduction and reactions with molecular neutrals are very small above 550 km. Under steady state conditions $\frac{\partial N_i}{\partial t} = 0$ as well, so with symmetric conditions, $\phi_{H^+}(S) + \phi_{O^+}(S) = 0$ by conservation of charge. In a separate work (Young et al., 1979b) have shown that quite simply, counterstreaming is required by the continuity equations. The particular direction of the counterstreaming

being determined by the relative densities of neutral H and neutral O at higher and lower altitudes. The counterstreaming of H^+ and O^+ is a fundamental result for steady state symmetric simulations with closed magnetic flux tube models which do not allow any cross field movement of ions.

Under time dependent simulations the condition is weakened to an equivalent statement about average fluxes (Young et al., 1979b).

$$<\phi_{H^+}(S)> = - <\phi_{O^+}(S)> \tag{17}$$

And in non symmetric time dependent cases the locally valid equation is further weakened to a statement concerning flux at both ends of a bounded portion of a field line (Young et al., 1979b).

$$(<\phi_{H^+}(b)> - <\phi_{H^+}(a)>) = - (<\phi_{O^+}(b)> - <\phi_{O^+}(a)>) \tag{18}$$

where a and b are two separate points on the field line.

Although the counterstreaming of H^+ and O^+ is a fundamental result applying to any model of a closed plasmaspheric flux tube allowing field aligned transport only, the effect has been widely overlooked in the literature because it is masked for the most part by diurnal variations. Time averaging would be needed to deconvolve the counterstreaming under realistic conditions. Notable exceptions in modeling (Bailey et al., 1977) and measurements (Vickery, 1979) have encountered counterstreaming only during short periods of real or simulated time in twilight conditions when the common diurnal component of ion flow was on the point of reversing itself. Careful measurement of the amount by which (17) and (18) do not hold true would be a measure of the total amount of cross field ion diffusion which will eventually be needed in a truly accurate model of the plasmasphere (Young et al., 1979b).

We have also simulated an ionospheric collapse, as might occur at sunset. For this purpose we allowed the electron heating rate and the photoproduction rate to decay exponentially at each point with a time constant of 20 minutes. The results are shown in Figure 5 as curves annotated with the time after the initiation of collapse. Times of 0h, 1h, 3h and 10h are shown for O^+, while only 0h and 10h are shown, for clarity of display, in the case of H^+.

The collapse of the topside O^+ density profile is the most profound sunset effect, while the topside H^+ density profile only appears to settle slightly. Both effects however are due to the same cause: the reduction of the topside ion and electron temperatures. In our temperature model, the equatorial temperatures are both reduced to about 1100° after 10 hours from initial values of T_i = 4500°K and T_e = 5500°K used for the steady state. The concomitant reduction in topside scale heights causes the O^+ density which had a shorter scale height to begin with to decrease more rapidly.

It will be noticed that the reduction of densities is moderate at middle altitudes and then again very pronounced near the lower boundary of our simulation. Temperature effects have less influence at low altitudes. The more dramatic decrease there is to be expected however, because the chemical destruction rate is much greater than at the middle altitudes.

The counterstreaming reported for steady state conditions disappears, to be replaced within about one hour by downstreaming of both ions everywhere. Even after 10h however, there remains a residual pattern that looks as if the steady state counterstreaming were superimposed upon a net downflux. It would appear that the profiles are still in the process of adjusting to the reduced temperatures. The reduced photoproduction can no longer support an upflux of O^+ at high altitudes in any case.

The H^+-O^+ transition height is also lowered at night. Where it was 1150 km under daytime steady state conditions it drops to only 650 km after 10h. The lower transition height is a combined effect resulting from the reduction of both the neutral and ion scale heights under the lower temperature post collapse conditions. Note that the H^+ density actually increases at around 800 km. The basic features agree with Evans' and Holts' (1978) geophysical observations and demonstrate clearly that the code is working properly.

Temperature Results

The model has also been used to simulate experimental data from the S3-3 satellite, for orbit number 1035 (Rich et al., 1978). A comparison between the theoretical electron temperature and measurements is shown in Figure 6.

The most significant feature of the comparison is the good agreement between the theoretical and experimental electron temperature gradients. Previous measurements have indicated the existence of large temperature gradients in the magnetosphere (Brace, 1970; Serbu and Maier, 1966). Such large gradients are inconsistent with the classical thermal conductivity of Spitzer, which has been used in our model, and which produces small gradients at high altitudes. Mayr and Volland (1968) have shown that outside the plasmapause where densities are very small, the thermal heat flow is greatly reduced. On the other hand, Mayr et al. (1973) showed that within the plasmasphere the thermal conductivity is only slightly changed from the classical theory. We have chosen to use the Spitzer form of the thermal conductivity until further theoretical work produces a different result.

Although the theoretical temperatures in Figure 6 are within the error bounds of the S3-3 results, such high temperatures were only obtained by including an extra heat source in the protonosphere. For, even under the assumption that all photoelectrons leaving the ionosphere lose their heat in the protonosphere,

the plasmaspheric electron temperature was 400°K lower than the data. The need to include an extra heat source may be due to the calculated photoelectron fluxes being too small. Comparison of theoretical and measured photoelectron fluxes seem to indicate that the theoretical fluxes may be too low by a factor of 2. However, the question is not yet resolved, and there is still a possibility that an extra heat source, such as wave-particle interactions may need to be included.

A further indication that the classical theory is adequate, is that the model not only reproduces high altitude temperatures, but low altitude temperatures compare favorably with typical measurements. This can bee seen in Figure 7 where the low altitude theoretical temperatures are plotted alongside 1964 Millstone Hill data (Evans, 1967). Although the theoretical temperatures are slighly higher than the experimental temperatures, they fall well within the day to day range of variability indicated by the data.

The time-dependent capabilities of the program are illustrated in Figure 8. Starting with the conditions indicated by Figure 6, the heating rate was reduced to simulate the decay of electron temperature at night. During the day, temperature changes are small, but during the night the temperature decays very rapidly at low altitudes and less rapidly at high altitudes where heat conduction is important and local heat loss is small.

It can be seen from Figure 8, that although $T_e > T_i$ during the day, at night $T_i > T_e$ after less than an hour, because electrons lose heat by conduction faster than ions. (The curves in Figure 8 represent hourly intervals). Figure 8 also shows that the rate of decay of electron temperature decreases with time as would be expected because of the high temperature dependence of the thermal conductivity. The temperature results shown here are similar to those obtained by Richards and Cole (1979) using the searching procedure. However, the above results are more accurate because the solution of the equations is simultaneous,

and because the whole field line has been simulated rather than assuming zero heat flux at the equatorial plane.

Comparison of Model with AE Data

Figure 9 gives a comparison between AE O^+ density data and that calculated from the model. The data were taken from orbit 2758 of the AE-C satellite, near 12.77 hrs UT.

Agreement in the altitude range 200-400 km is good but above 400 km agreement is very poor. Comparison between satellite data and the model is complicated by the latitudinal variation of the data. During this pass the high altitude data was taken near L = 3.5 while the low altitude data was taken near L = 1.5. The differences between data and theory could be due to latitudinal gradients of density.

The electron and ion temperatures are compared in Figure 10. The agreement in this case is much better than for the density. The disagreement between the measured and theoretical ion temperatures below 450 km indicates that the MSIS model neutral temperature may be too low for this particular day.

2.7 Future Improvements to the Simulation

Our simulated density profiles show that our approximation of dynamic equilibrium is satisfactory for H^+ and O^+ above the altitude where H^+ becomes the only major ion. Nevertheless there is an almost imperceptible discontinuity in the density gradients at boundaries 2 and 2*, so we intend to include ion-ion collisions at the very high altitudes at a later date. Two possible ways to do this present themselves. One is to formulate O^+ density as a parabolic equation at all altitudes (regions B, C and B*), but retain first order momentum and continuity equations for H^+ at high altitudes (region C). Another way is to upgrade the dynamic equilibrium solutions iteratively by calculating each V_i from the continuity equation integrated between boundary 2 and 2*, and then substitute it into the collision terms of the full momentum equation

$$V_i(S) = \frac{B(S)}{N_i(S)} \left\{ \frac{\Phi_i(2)}{B(2)} + \int_2^S \left[P_i - \hat{L}_{ij} N_j - \frac{\partial N_i}{\partial t} \right] \frac{ds}{B(S)} \right\} \quad (19)$$

$$Q_i = - \frac{1}{D_i} \left[(V_i - V_j) + \frac{\nu_{ix}}{\nu_{ij}} V_i \right] \quad (20)$$

The velocity V_i in equation (21) would be calculated from the previous iteration. Note that Q_i will now be equal to some small correction number, rather than identically equal to zero, as used in the dynamic equilibrium approximation.

The code will be used to study a number of geophysical problems including diurnal variations under a number of geophysical conditions, effects of inter-hemispheric flows, magnetospheric substorms and seasonal and solar cyclic variations upon profiles, flux tube content and filling processes. Future simulations will be compared extensively with satellite and ground based measurements. The electron and ion energy equations will be simulated simultaneously as well. Future runs should therefore shed considerable light upon diurnal variations, plasmaspheric heat sources, ion fluxes and the maintenance of the nighttime ionosphere.

2.8 Conclusions

We have achieved the most comprehensive simulation to date of field aligned plasma transport in the plasmasphere. The plasma simulation itself incorporates the best aspects of two older methods in a unified mathematical model and uses accurate geophysical parameters to produce an accurate and meaningful solution to the geophysical plasma problem.

We have shown that we can link the solutions for an entire flux tube connecting the ionosphere and protonosphere even though it passes through several regions that require different sets of differential equations to approximate conditions in different regimes. We have derived the continuity relations across the boundary regions between different approximations, necessary for this type of approach, which must replace artificially specified boundary conditions used in previous work. We have shown that a high altitude counter-streaming of H^+ and O^+ must and does result under steady state conditions.

Acknowledgments

The authors gratefully acknowledge helpful discussions with A. F. Nagy, R. W. Schunk, J. P. St. Maurice and A. E. Hedin. This work was supported by AF contract F19628-77-C-0007. Computing support was provided by the National Center for Atmospheric Research, which is sponsored by the National Science Foundation.

2.9 References

Abur-Robb, M. F. K., Planet. Space Sci., 17, 1269-1279, 1969.

Bailey, G. J., R. J. Moffett and H. Rishbeth, J. Atmos. Terr. Phys., 31, 253-270, 1969.

Bailey, G. J., R. J. Moffett and J. A. Murphy, Planet. Space Sci., 25, 967-972, 1977.

Bailey, G. J., R. J. Moffett and J. A. Murphy, Planet. Space Sci., 26, 753-765, 1978.

Banks, P. M., Behavior of thermal plasma in the magnetosphere and topside ionosphere, in: Critical Problems of Magnetospheric Physics, Proc. of the Joint COSPAR/IAGA/URSI Symposium, 1972.

Banks, P. M. and A. F. Nagy, J. Geophys. Res., 75, 1902, 1970.

Banks, P. M., A. F. Nagy and W. I. Axford, Planet. Space Sci., 19, 1053, 1971.

Banks, P. M., R. W. Schunk and W. J. Raitt, The topside ionosphere: A region of dynamic transition, in Annual Review of Earth and Planetary Sciences, 4, 1976.

Bauer, P., The cooling of the protonosphere, Ph.D. Thesis, University of Michigan, 1968.

Bauer, S. J., The structure of the topside ionosphere, in Electron Density Profiles in the Ionosphere and Exosphere (Ed. J. Frihagn), North-Holland, Amsterdam, 1966.

Behnke, R. A., Vector measurements of the ion transport velocity with applications to F region dynamics, Ph.D. Thesis, Rice University, Houston, Texas, 1979.

Behnke, R. and H. Kohl, J. Atmos. Terr. Phys., 36, 325-333, 1974.

Blum, P. W. and I. Harris, J. Atmos. and Terr. Phys., 37, 213, 1975.

Brace, L. H., The global structure of ionosphere temperature, Space Res. X, T. M. Donahue, P. A. Smith, L. Thomas, eds North-Holland Publishing Co., Amsterdam, 633, 1970.

Brace, L. H., E. J. R. Maier, J. H. Hoffman, J. Whitteker and G. G. Shepherd, J. Geophys. Res., 79, 5211, 1974.

Brace, L. H., B. M. Reddy and H. G. Mayr, J. Geophys. Res., 72, 265, 1967.

Burgers, J. M., Flow equations for composite gases, Academic Press, New York, 1969.

Carpenter, D. L. and C. G. Park, Rev. Geophys. Space Phys., 11, 133, 1973.

Carpenter, D. L. and R. L. Smith, Rev. Geophys. Space Phys., 2, 415, 1964.

Chapman, S. and T. G. Cowling, The mathematical theory of non-uniform gases, Cambridge University Press, Cambridge, England, 1970.

Chappell, C. R., Rev. Geophys Space Phys., 10, 951, 1972.

Chappell, C. R., Rev. Geophys. Space Phys., 872-873, 1975.

Evans, J. V., Planet. Space Sci., 13, 1387, 1967.

Evans, J. V., Theory and practice of ionosphere study by Thomson scatter radar, Proc. IEEE, 57, 496, 1969.

Evans, J. V., Radio Sci., 6, 609, 1971a.

Evans, J. V., Radio Sci., 6, 843, 1971b.

Evans, J. V., Rev. Geophys. Space Phys., 887, 1975a.

Evans, J. V., Planet. Space Sci., 23, 1461-1482, 1975b.

Evans, J. V., R. F. Julian and W. A. Reid, Incoherent scatter measurements of F region density temperatures and vertical velocity at Millstone Hill, M.I.T. Lincoln Lab. Rept. 477, 1970.

Fontheim, E. G. and P. M. Banks, Planet. Space Sci., 20, 73, 1972.

Geisler, J. E., J. Geophys. Res., 72, 81-85, 1967.

Geisler, J. E. and S. A. Bowhill, An investigation of ionosphere-protonosphere coupling, Aeronomy Rept. 5, University of Illinois, Urbana, 1965.

Grad, H., Commun. Pure Appl. Math., 2, 331, 1949.

Grad, H., Handb. Phys., 12, 205, 1958.

Hagen, J. B., A hybrid autocorrelator and its application to high altitude incoherent scatter, Nat. Astr. Ionosphere Centre, Arecibo, Puerto Rico Rept. 17, 1972.

Hanson, W. B., Dynamic diffusion processes in the exosphere, in Electron Density Profiles in the Ionosphere and Exosphere, p. 361, North-Holland Amsterdam, 1964.

Hanson, W. B. and T. N. L. Patterson, Planet. Space Sci., 12, 979-997, 1964.

Hedin, A. E., M. E. Salah, J. V. Evans, C. A. Reber, G. P. Newton, N. W. Spencer, D. C. Kayser, D. Alcayde, P. Bauer, L. Cogger and P. McClure, J. Geophys. Res., 82, 2139, 1977a.

Hedin, A. E., C. A. Reber, G. P. Newton, N. W. Spencer, H. C. Brinton, H. G. Mayr and W. E. Potter, J. Geophys. Res., 82, 2148, 1977b.

Hirschfelder, J. O., C. F. Curtis and R. B. Bird, Molecular theory of gases and liquids, John Wiley, New York, 1964.

Johnson, F. S., J. Geophys. Res., 65, 577, 1960.

Jones, K. L., J. Atmos. Terr. Phys., 36, 397-406, 1974.

Kockarts, G. and M. Nicolet, Ann. Geophys., 19, 370, 1963.

Kohl, H., The possible effect of diffusion between magnetically conjugate points on the seasonal anomaly of the F layer, in Electron Density Profiles in the Ionosphere and Exosphere, p. 231, North-Holland, Amsterdam, 1966.

Kohl, H., J. W. King and D. Eccles, J. Atmos. Terr. Phys., 30, 1733-1744, 1968.

Kutimskaya, N. A., V. M. Polyakov, N. N. Klimov, C. M. Kuznetosova and G. I. Gerschengorn, Geomag. and Aeron., 13, 33, 1973.

Mahajan and Brace, L. H., J. Geophys. Res., 74, 5099, 1969.

Maher, L. J. and B. A. Tinsley, J. Geophys. Res., 82, 689, 1977.

Mange, P., J. Geophys. Res., 65, 3833, 1960.

Massa, J. L., Theoretical and experimental studies of the ionization exchange between the ionosphere and plasmasphere, Ph.D. Thesis, University of Michigan, 1974.

Massa, J. L., R. J. Cicerone and A. F. Nagy, Theoretical and experimental studies of ionization exchange between the ionosphere and plasmasphere, Paper presented at the Int. Symp. Solar Terrestrial Physics, Sao Paulo, Brazil, Summary in Atmosphere 3, 338, 1974.

Mayr, H. G., L. H. Brace and G. S. Dunham, J. Geophys. Res., 72, 4391, 1967.

Mayr, H. G., E. G. Fontheim, L. H. Brace, H. C. Brinton and H. A. Taylor, J. Atmos. Terr. Phys., 34, 1659, 1972.

Mayr, H. G., E. G. Fontheim and K. K. Mahajan, Ann. Geophys., 29, 21, 1973.

Moffett, R. J. and J. A. Murphy, Planet. Space Sci., 21, 43, 1973.

Murphy, J. A. and R. J. Moffett, Planet. Space Sci., 26, 281, 1978.

Murphy, J. A., G. J. Bailey and R. J. Moffett, J. Atmos. Terr. Phys., 38, 351, 1976.

Nagy, A. F. and P. M. Banks, J. Geophys. Res., 77, 4277-4279, 1972.

Nagy, A. F., P. Bauer and E. G. Fontheim, J. Geophys. Res., 73, 6259, 1968.

Park, C. G. and P. M. Banks, J. Geophys. Res., 79, 4661-4668, 1974.

Rich, F. J., R. C. Sagalyn and P. J. L. Wildman, J. Geophys. Res., 84, 1328, 1979.

Richards, P. G., A numerical study of ionosphere-protonosphere coupling, Ph.D. Thesis, LaTrobe University, Melbourne, Australia, 1978.

Richards, P. G. and K. D. Cole, Planet. Space Sci., in press, 1979.

Rishbeth, H., Proc. IEEE, 55, 16-35, 1967.

Rishbeth, H., J. Atmos. Terr. Phys., 30, 63-71, 1968.

Rishbeth, H., J. Atmos. Terr. Phys., 34, 1-47, 1972.

Rishbeth, H., J. Atmos. Terr. Phys., 36, 2309-2319, 1976.

Roble, R. G., Planet. Space Sci., 23, 1017-1033, 1975.

Rothwell, P., Diffusion of ions between F layers at magnetic conjugate points, Proc. Inst. Conf. Ionosphere, p. 217, Inst. Phys. and Phys. Soc., London, 1963.

Sanatani, S. and W. B. Hanson, J. Geophys. Res., 75, 769, 1970.

Schunk, R. W., Planet. Space Sci., 23, 437-485, 1975.

Schunk, R. W., Rev. Geophys. Space Phys., 15, 429, 1977.

Schunk, R. W. and J. C. G. Walker, Planet. Space Sci., 17, 853, 1969.

Schunk, R. W. and J. C. G. Walker, Planet. Space Sci., 18, 535, 1970a.

Schunk, R. W. and J. C. G. Walker, Planet. Space Sci., 18, 1319, 1970b.

Schunk, R. W. and J. C. G. Walker, Planet. Space Sci., 20, 581-589, 1972.

Schunk, R. W. and J. C. G. Walker, Planet. Space Sci., 21, 1875, 1973.

Schunk, R. W. and D. S. Watkins, Planet. Space Sci., 27, 433, 1979.

Serbu, G. P. and E. J. Maier, Space Res. VII, 527, 1967.

St. Maurice, J. P. and R. W. Schunk, Planet. Space Sci., 25, 907-920, 1977.

Sterling, D. L., W. B. Hanson, R. J. Moffett and R. G. Baxter, Radio Sci., 4, 1005-1023, 1969.

Strobel, D. G. and M. B. McElroy, Planet. Space Sci., 18, 1181-1202, 1970.

Stubbe, P., J. Atmos. Terr. Phys., 30, 243-263, 1968.

Stubbe, P., J. Atmos. Terr. Phys., 32, 865-903, 1970.

Vasseur, G. and P. Waldtenfel, J. Atmos. Terr. Phys., 30, 779, 1968.

Vickery, J. F., W. E. Swartz and D. T. Farley, Geophys. Res. Lett., 3, 217, 1976.

Walker, J. C. G., Planet. Space Sci., 15, 1151, 1967.

Young, E. R., P. G. Richards and D. G. Torr, submitted to J. Comp. Phys., 1979a.

Young, E. R., P. G. Richards and D. G. Torr, submitted to J. Geophys. Res., 1979b.

Figure Captions

Figure 1. Regions and Boundaries.

A, A*	Regions of local chemistry
B, B*	Ion diffusion regions where parabolic density equations can be formulated
C	Dynamic equilibrium region
1, 1*, 2, 2*	Boundaries between adjacent regions: at 1 and 1* densities are continuous, while at 2 and 2* both density and flux are continuous

Figure 2 Overall Schematic.

The symbols on the diagram are

T_e = electron temperature

T_i = ion temperature

T_n = neutral temperature

N_i = ion density

N_n = neutral density

ϕ_e = electron flux

ϕ_i = ion flux

Ion Con = Ion continuity equations

Ion Mom = Ion momentum equations

The high altitude regime is denoted "Dynamic Equilibrium" while the low attitude regime is denoted parabolic EQ.

Figure 3. Schematic of spatial grid.

It is to be noticed the boundaries 2 and 2* are each straddled by the pairs of points (2A, 2B) and (2A*, 2B*) respectively, from which the boundary fluxes are calculated. The boundaries 1 and 1* on the other hand fall exactly upon the first and last point. The densities at these points are calculated via local photochemical equilibrium. Both densities at each of 3 points are needed to calculate each finite difference equation used in the Newton Raphson Iterative solution. The three points are adjacent for equation 12 but not for equation 13. The three points needed to calculate equation 13 are 2B, 2A and 2B*.

Figure 4. Simulated Ion Fluxes.

Both H^+ and O^+ fluxes are shown, Oh denoting steady state collisions, and 10h the ion fluxes 10h after a simulated ionospheric collapse. For steady state conditions, denoted Oh in the figure, the H^+ flux is downward everywhere while the O^+ flux is divided into a downward regime below 500 km and an upward regime above. It is apparent that above about 600 km, where $O^+ + H \quad H^+ + O$ are the only chemical reactions the two fluxes are virtually equal and opposite. Another striking feature, comparing with Figure 4 is that the maximum O^+ flux occurs right at the maximum in the O^+ density profile.

Ten hours after the collapse, denoted 10h, both fluxes are everywhere downward. The O^+ downflux is higher than previously at high altitudes, but lower below 350 km because of a decrease in ion density.

Figure 5. Simulated Ion Densities.

For H^+, curves for steady state conditions and ten hours after the collapse denoted Oh and 10h respectively, are shown. For O^+ the curves denoted Oh, 1h, 3h and 10h denote steady state and then one, three and ten hours after the simulated ionospheric collapse.

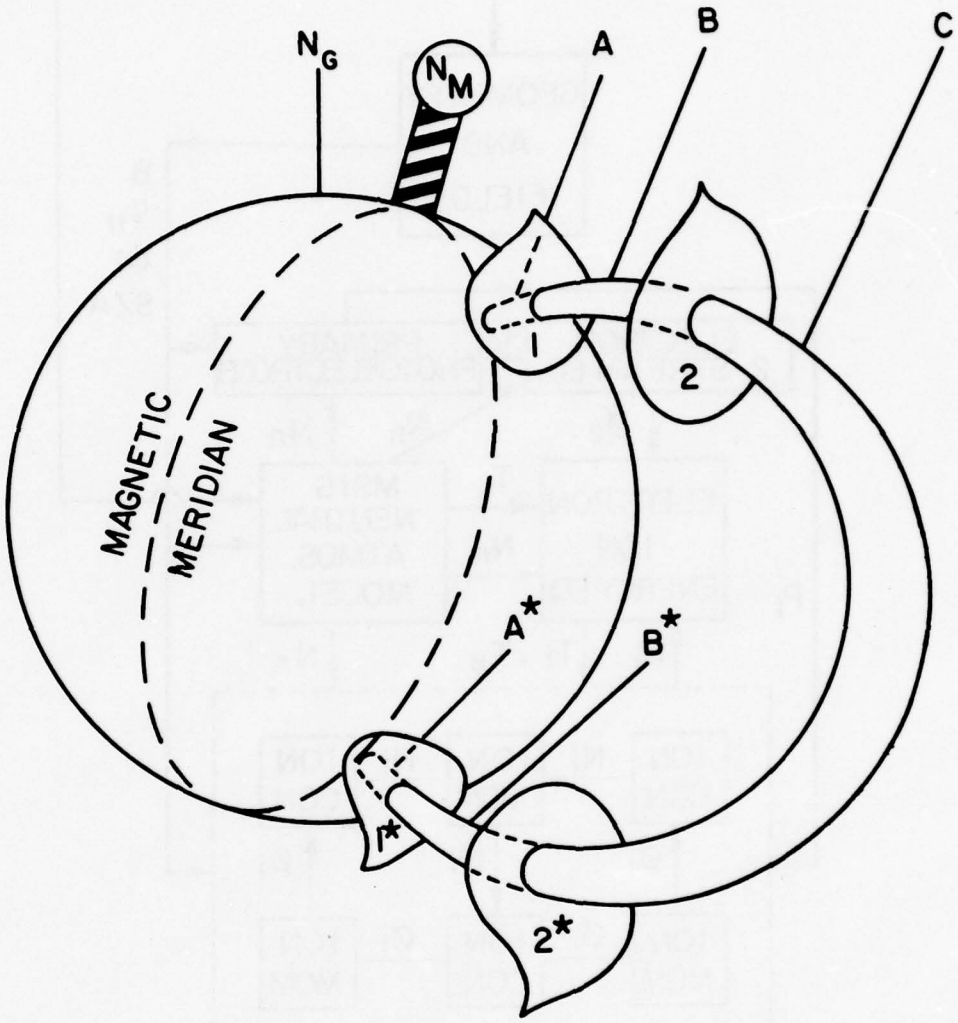
Figure 6. The results of a theoretical simulation of the S3-3 satellite measurements for orbit #1035. The theoretical results have been obtained through a steady state solution of the energy equations. An extra heat source was needed to obtain comparable temperatures.

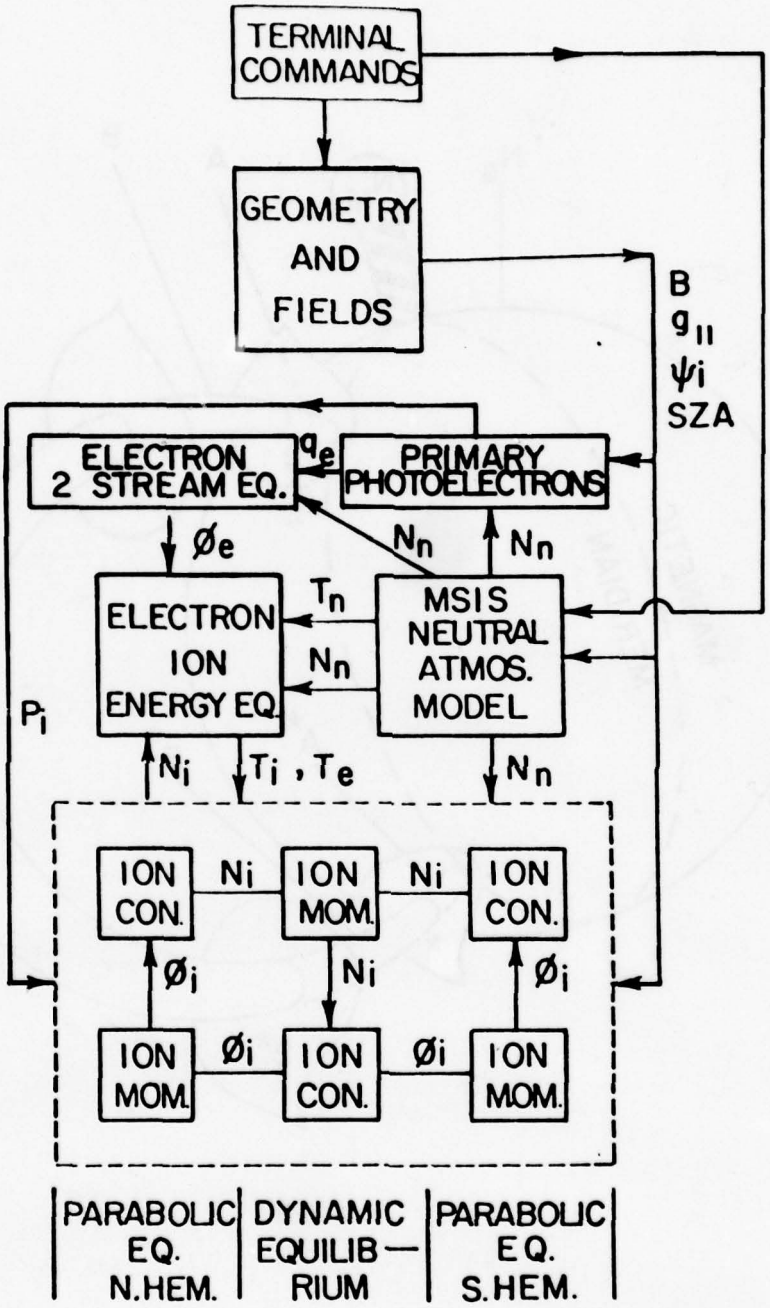
Figure 7. Comparison of theoretical results with the data of Evans (1967) at low altitudes. The lower temperatures of the data indicates that the plasmaspheric temperature was lower in 1964 than that measured by S3-3 in 1976.

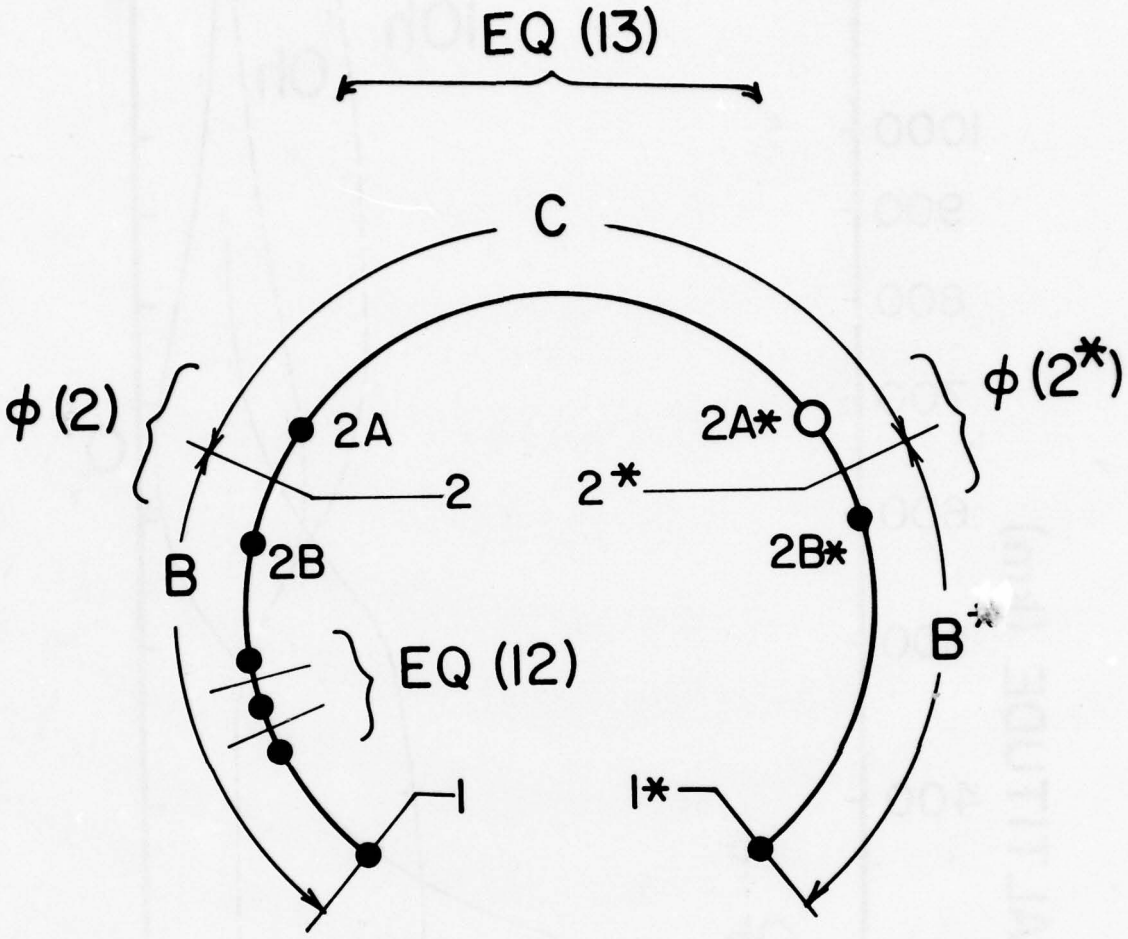
Figure 8. Ion and electron temperature decay, starting with the conditions of Figure 1 and switching off the heating source. The curves are given at hourly intervals.

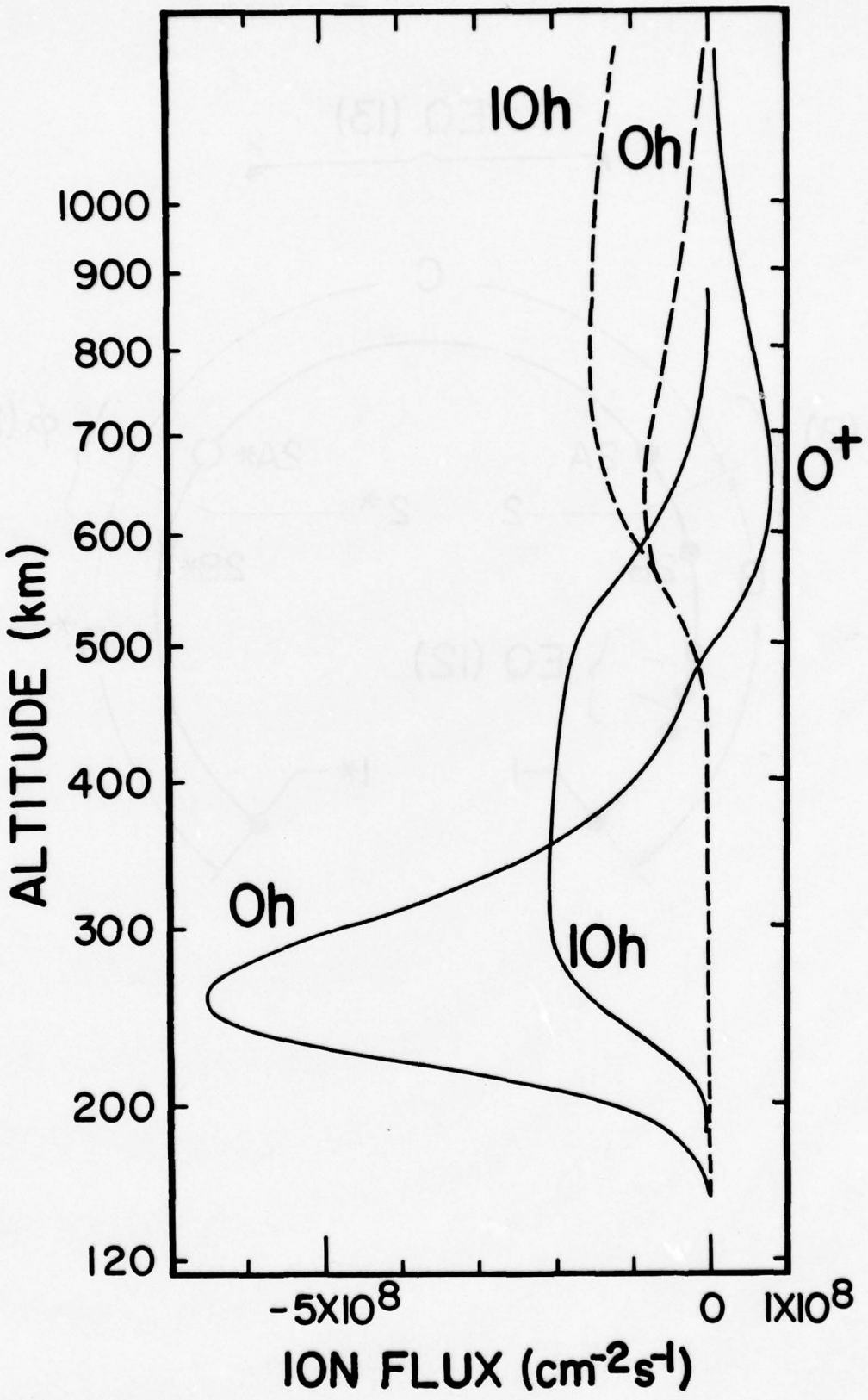
Figure 9. Comparison between model and AE-C O^+ densities for AE-C orbit number 2758.

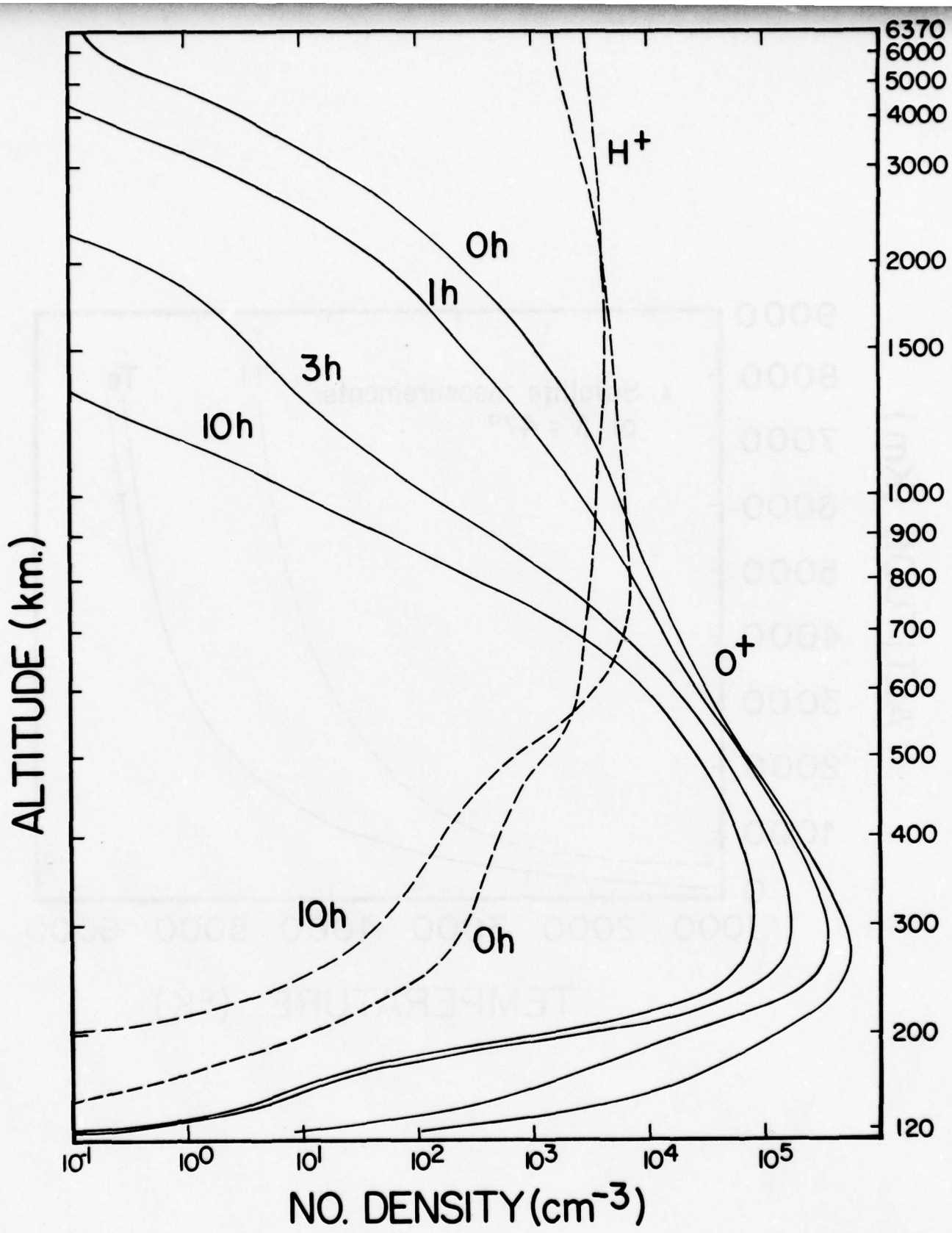
Figure 10. Comparison between model and AE-C ion and electron temperatures for AE-C orbit number 2758.

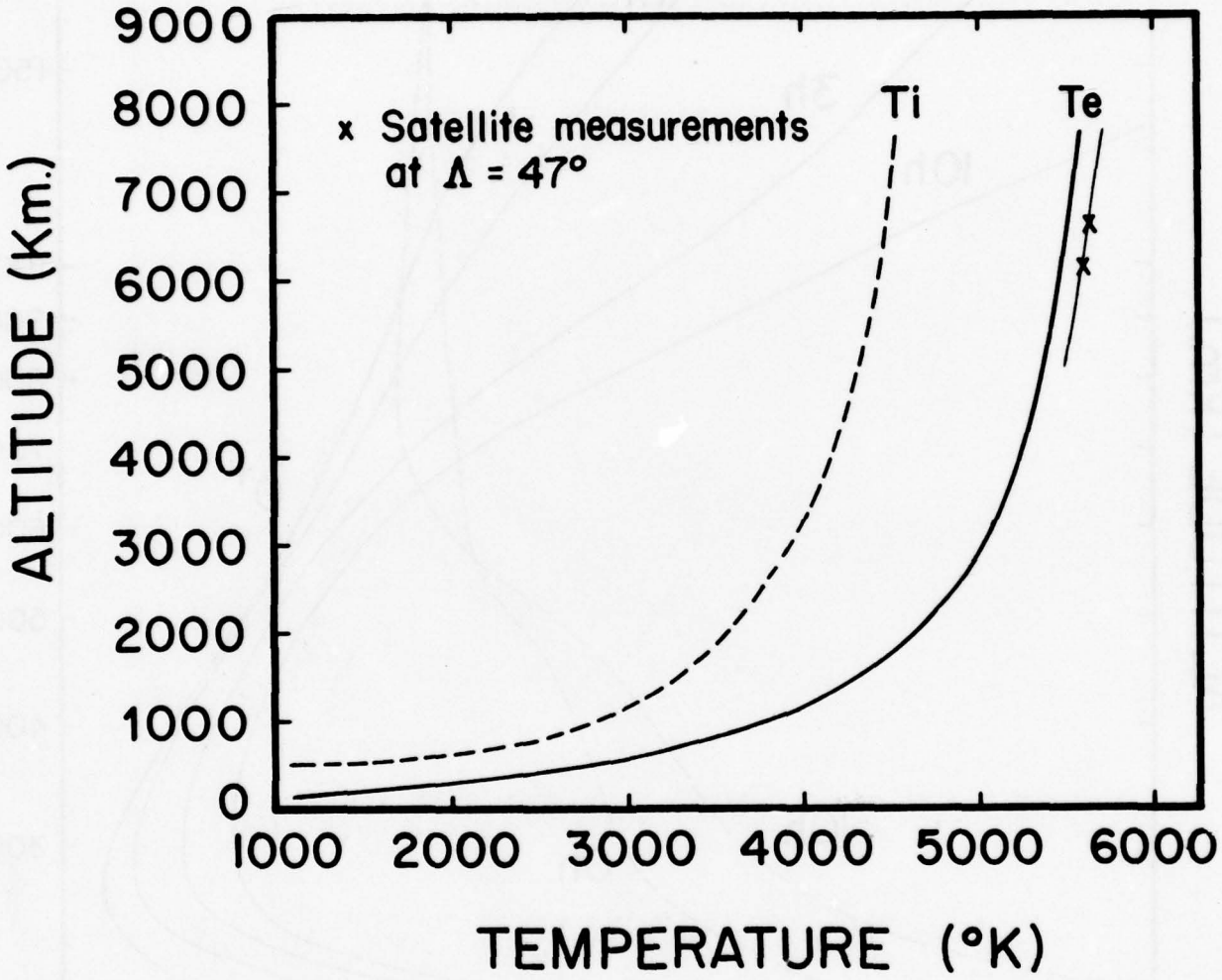

N_G
N_M
A
B
C
MAGNETIC MERIDIAN
1
2
A*
B*
1*
2*

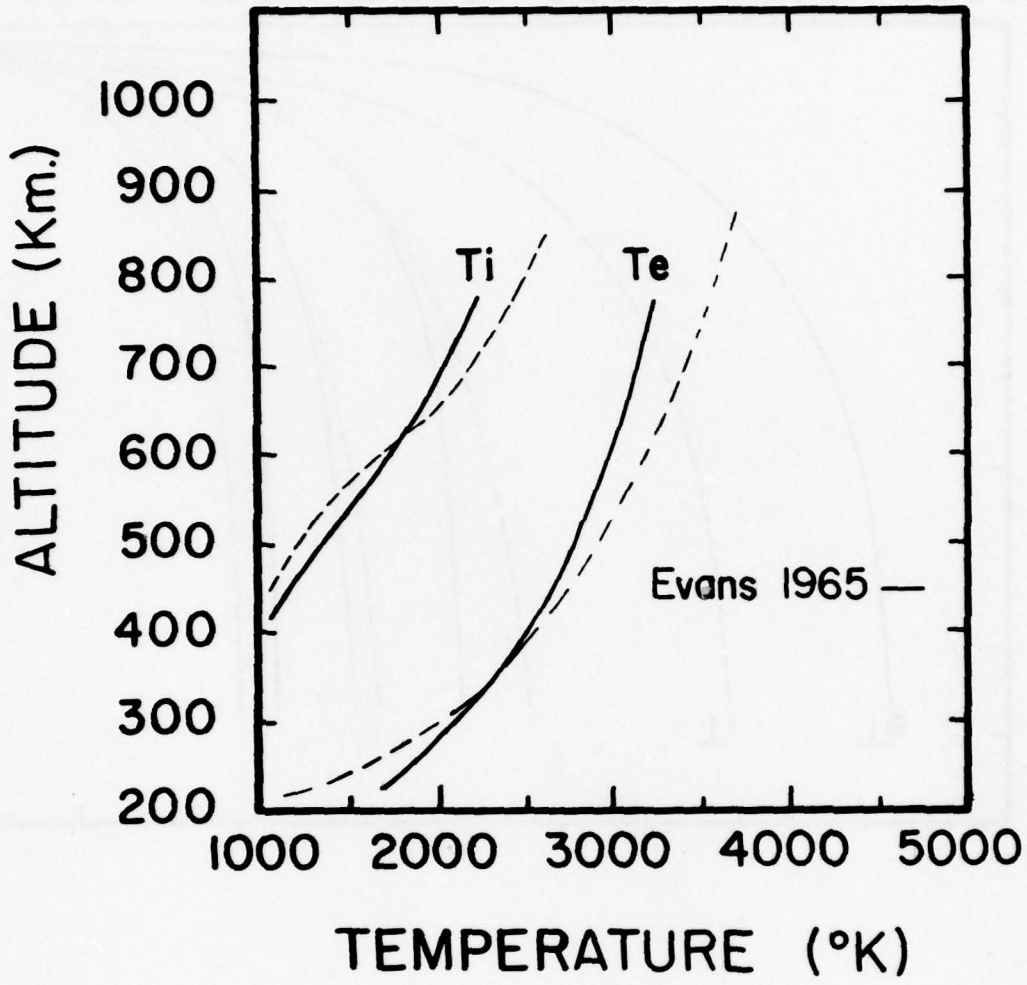

TERMINAL COMMANDS
GEOMETRY AND FIELDS
B
g_{11}
ψ_i
SZA
ELECTRON 2 STREAM EQ.
q_e
PRIMARY PHOTOELECTRONS
ϕ_e
N_n
N_n
ELECTRON ION ENERGY EQ.
T_n
N_n
MSIS NEUTRAL ATMOS. MODEL
P_i
N_i
T_i , T_e
N_n
ION CON.
N_i
ION MOM.
N_i
ION CON.
ϕ_i
N_i
ϕ_i
ION MOM.
ϕ_i
ION CON.
ϕ_i
ION MOM.
PARABOLIC EQ. N.HEM.
DYNAMIC EQUILIB— RIUM
PARABOLIC EQ. S.HEM.

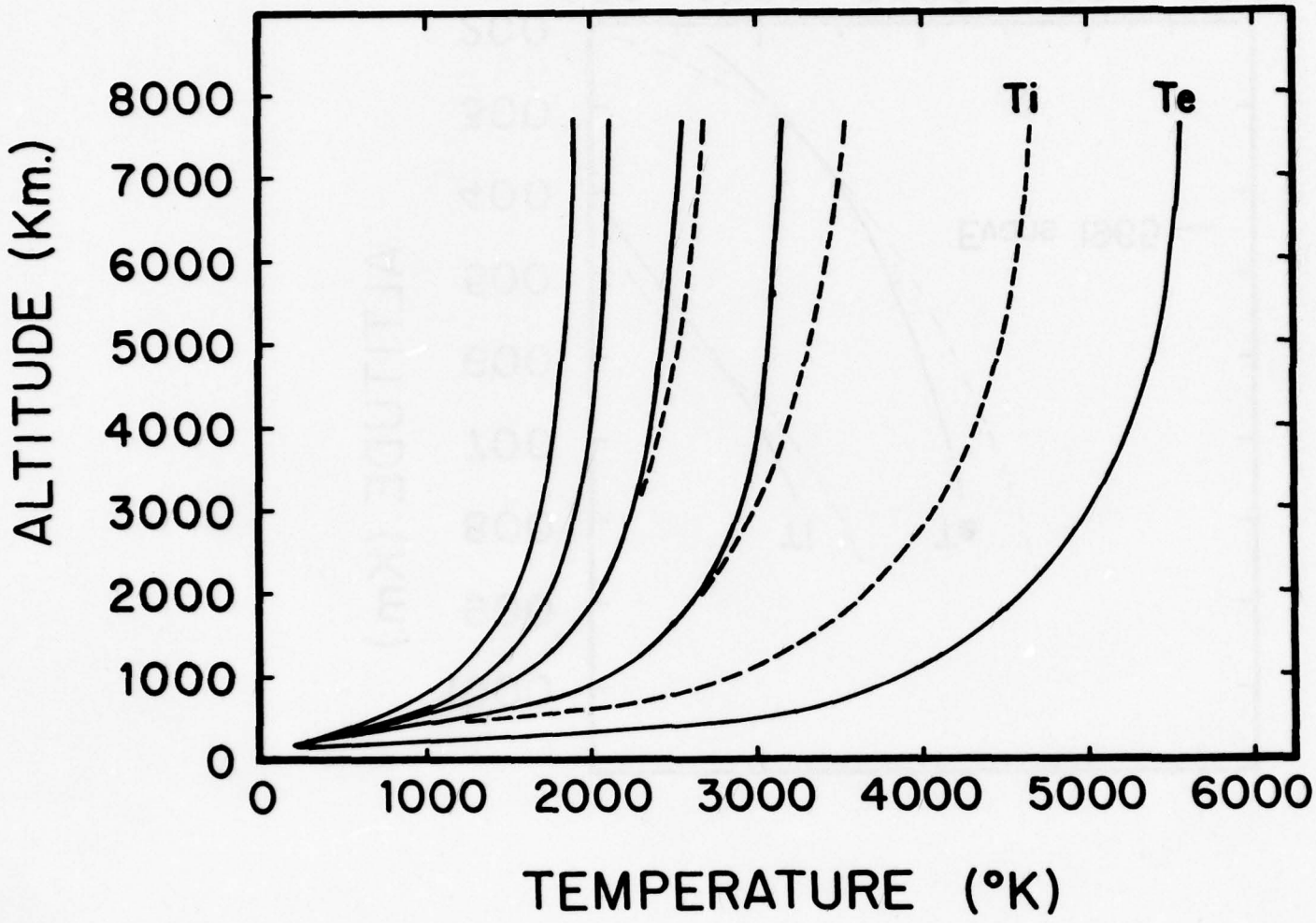

EQ (13)
C
φ (2)
φ (2*)
2A
2A*
2
2*
B
2B
2B*
B*
EQ (12)
1
1*

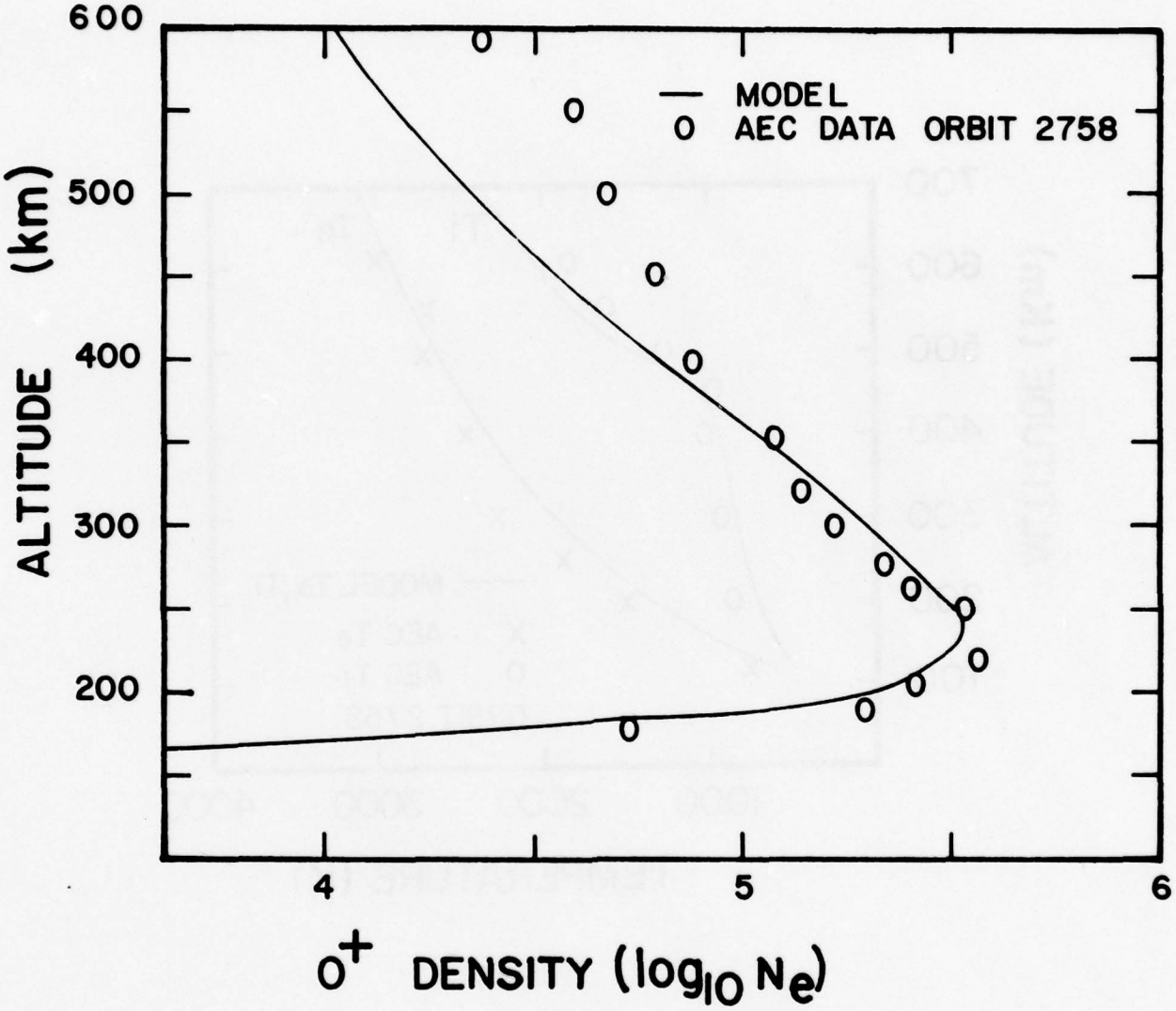

10h
0h
1000
900
800
700
600
500
400
300
200
120
ALTITUDE (km)
O+
0h
10h
-5X10^8
0
1X10^8
ION FLUX (cm^-2 s^-1)

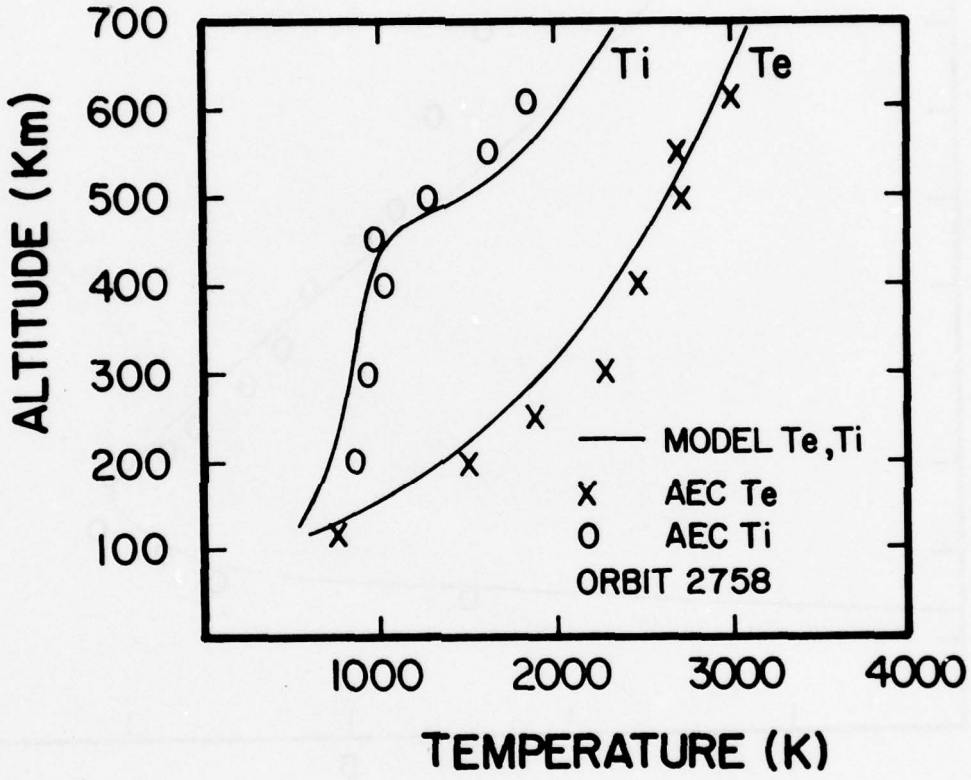

6370
6000
5000
4000
3000
2000
1500
1000
900
800
700
600
500
400
300
200
120
H+
0h
1h
3h
10h
O+
10h
0h
ALTITUDE (km.)
10^-1
10^0
10^1
10^2
10^3
10^4
10^5
NO. DENSITY (cm^-3)


9000
8000
7000
6000
5000
4000
3000
2000
1000
0
ALTITUDE (Km.)
x Satellite measurements
at Λ = 47°
Ti
Te
1000 2000 3000 4000 5000 6000
TEMPERATURE (°K)


ALTITUDE (Km.)
1000
900
800
700
600
500
400
300
200
Ti
Te
Evans 1965
1000
2000
3000
4000
5000
TEMPERATURE (°K)


ALTITUDE (Km.)
8000
7000
6000
5000
4000
3000
2000
1000
0
0
1000
2000
3000
4000
5000
6000
TEMPERATURE (°K)
Ti
Te


600
500
400
300
200
ALTITUDE (km)
MODEL
AEC DATA ORBIT 2758
4
5
6
O+ DENSITY (log10 Ne)


700
600
500
400
300
200
100
ALTITUDE (Km)
Ti
Te
MODEL Te,Ti
X AEC Te
O AEC Ti
ORBIT 2758
1000
2000
3000
4000
TEMPERATURE (K)

3.0 APPENDICES

3.1 SUMMARY OF CONTRACT OBJECTIVES

This study falls under the general category of "Plasma exchange between the ionosphere and magnetosphere". We have contracted to carry out the following work.

a) Solve general transport equations for the F region and topside ionosphere. Solutions will subsequently be compared with those derived from conventional measurements including results from plasma measurements made by Injun V, ISIS-I and AF S3-2 and S3-3 satellites. The sensitivity of these solutions to varying geophysical conditions will be assessed by appropriate variation of parameters and where practical analytical expressions will be derived for parameters including drift velocity, and thermal diffusion in a multicomponent plasma.

b) Compare solution of the transport equations for O^+, H^+ and electrons with data obtained from the Arecibo Observatory sounder. These computations will be made for days when a full diurnal cycle of operation at Arecibo and simultaneous neutral densities from Atmosphere Explorer satellites are available.

c) Extend studies to include effects of thermal diffusion and thermoelectric effects. Electron and ion temperature will be computed along magnetic field lines extending into the conjugate region, and the results applied to study of downward flux of plasma in nighttime F region, filling of depleted field tubes after magnetic storms, and the discrepancy noted by Massa between calculated and observed H^+ densities for Millstone Hill incoherent scatter facility. The low altitude effect of neutral winds on these processes will be studied.

d) Give examples of the computer codes employed in the work described in paragraphs 2.0 a, b, and c of this report.

3.2 CUMMULATIVE CHRONOLOGICAL NARRATIVE

During the first quarter, Drs. J. P. St. Maurice and R. W. Schunk derived a set of two ion diffusion and heat flow equations for application to the midlatitude topside ionosphere. In this approach the velocity distributions are expanded about the drift velocity of each component. Several new transport effects not evidenct in the classical Chapman-Ensog formulation were discovered as a result of this new approach.

During the second quarter the diffusion and thermal conduction equations Computer routines were developed for the inversion of both stepped band matrixes and general sparse matrixes. The stepped band matrix inverter is used in the current code. It was decided that chemical rate coefficients for reactions of O^+ with neutrals should be determined from AE satellite measurements rather than lab data, because the conditions in the laboratory do not accurately represent the vibrational and speed distributions in the ionosphere.

In the third quarter, a numerical predictor corrector method was proposed and derived to replace the Newton iteration technique. This technique was not exploited in subsequent quarters, but could still prove valuable in future work.

In the fourth quarter, a number of programming and geophysical errors were eliminated from the code.

During the fifth quarter, the ion diffusion equations of St. Maurice and Schunk were reformulated slightly to include ion-neutral collisions. This allowed their substitution in the continuity equations which then became parabolic density equations. An innovative step was also taken by formally integrating these equations before applying finite differencing approximations. This technique automatically allowed the use of a variable spatial step size, and greatly facilitated the solution of the interhemispheric coupling problem.

Additional numerical problems were overcome during the sixth quarter. It was found that the determinant which arises in the solution of the simultaneous diffusion equations for O^+ and H^+ is the small difference of several large terms which nearly cancel each other. When two of the terms were cancelled analytically and the order of addition of the remaining terms optimized, the numerical accuracy and stability were greatly improved.

During the seventh quarter, the heat conduction and energy code became operational.

A photoelectron 2-stream program of Nagy and Banks (1970) was adopted to field line coordinates so that it could be extended beyond its original limited altitude range. This provided electron heating rates. Also the parabolic temperature equations were written as a variable step length function of dipole coordinates which greatly facilitated the treatment of the great distances involved in an entire field line. Upon comparison of our simulated results with the satellite measurements of Rich et al. (1978), it was found necessary to postulate an additional heat source to bring the simulated plasmaspheric temperatures up to those of the measurements. This remains a puzzle on lower L shells, but we are currently convinced that the results are satisfactory for L = 3 and compare well with observations from Millstone Hill.

During the eighth reporting period some remaining problems with high altitude convergence were solved by assuming a two ended diffusive equilibrium profile above about 1500 km. The derivation of proper boundary relations between 2 low altitude regions of parabolic equations and the high altitude region of diffusive equilibrium profiles is a crucial step which allowed the Newton iteration to work properly and simulate the density and momentum equations for an entire field line. Dr. Young presented two talks as follows on our new method and preliminary results.

(1) "Interhemispheric coupling of hydrogen and oxygen ions in the plasmasphere", Talk at the 1978 Fall Convention of the American Geophysical Union, 4 Dec. 1978, San Francisco.

(2) "Method and preliminary results of a simulation of a coupled ionosphere and plasmasphere", Geophysics Seminar, University of Washington, 8 Dec. 1978, Seattle, WA.

During the 9th reporting period additional geophysical simulations were performed. A journal article was submitted to Journal of Computational Physics detailing our innovative numerical methods of solving the continuity and momentum equations and analyzing fluxes and density profiles both in the steady state and under conditions simulating the post-sunset collapse of the ionosphere.

During the tenth and final reporting period the continuity, momentum, temperature and heat flow and photoelectron two-stream codes were finally working in combined form on the CRAY-1. With this combined code, which is described in the report on scientific work section of this report, we have simulated up to 10h of geophysical time. A paper is being prepared for J. Geophysical Research, but we are currently stalled because we have run out of time on the CRAY-1. CRAY-1 time is the only thing we need to finish the JGR paper.

3.3 LIST OF PERSONNEL CONNECTED WITH THIS CONTRACT

Name	Responsibility	Rank
Dr. D. G. Torr	Principal Investigator	Associate Research Scientist
Dr. J. P. St. Maurice	Co-Investigator	Post-doctoral Scholar
Dr. E. R. Young	Primary Researcher	Post-doctoral Scholar
Dr. P. G. Richards	Researcher	Post-doctoral Scholar
J. H. Waite	Researcher	Graduate Student Candidate
R. Gordon	Programming	Professional Programmer
S. Hsiung	Programming	Professional Programmer
Y. Huang	Programming	Graduate Student Assistant

3.4 LIST OF PUBLICATIONS

J. P. St. Maurice and R. W. Schunk, Diffusion and heat flow equations for the midlatitude topside ionosphere, Planet. Space Sci., 25, 907, 1977.

J. P. St. Maurice and D. G. Torr, Nonthermal rate coefficients in the ionosphere: The reactions of O^+ with N_2, O_2 and NO, J. Geophys. Res., 83, 969, 1978.

D. G. Torr and M. R. Torr, Review of rate coefficients of ionic reactions derived from measurements made by the Atmosphere Explorer satellites, Rev. Geophys. and Space Phys., 16, 327, 1978.

P. G. Richards, E. R. Young and D. G. Torr, Energy and heat flow in a plasmaspheric flux tube, Paper in preparation for Planetary and Space Sci.

E. R. Young, P. G. Richards and D. G. Torr, A flux preserving method of coupling first and second order equations to simulate the flow of plasma between the protonosphere and the ionosphere, submitted to J. Comp. Phys., 1979.

3.5 LIST OF SCIENTIFIC TALKS

J. P. St. Maurice and R. W. Schunk, Transport coefficients for the midlatitude topside ionosphere, paper presented at the Fall meeting of the AGU, Abstract in EOS, 57, December, 1976.

J. P. St. Maurice, D. G. Torr and M. R. Torr, The effect of an energy dependent cross section on the determination of some ion-neutral reaction rates in the ionosphere, paper presented at the Spring Meeting of the AGU, Abstract in EOS, 58, June 1977.

D. G. Torr and M. R. Torr, The F-region winter anomaly, Bryce Mt. AE Satellite Team Meeting, 1978.

E. R. Young and D. G. Torr, Progress report on a code to study an entire plasmaspheric flux tube, Talk to Plasma Science Group, Huntsville, Alabama August, 1978.

E. R. Young and D. G. Torr, Interhemispheric coupling of hydrogen and oxygen ions in the plasmasphere, Talk at the 1978 Fall Convention of the American Geophysical Union, 4 Dec. 1978, San Francisco.

E. R. Young and D. G. Torr, Method and preliminary results of a simulation of a coupled ionosphere and plasmasphere, Geophysics Seminar, University of Washington, 8 Dec. 1978, Seattle, WA.

M. R. Torr and D. G. Torr, Transport of doubly charged oxygen ions in the thermosphere, 1979 Spring AGU Meeting, Washington, D.C.

```
IT,D=PLIB,DN=RSMN,DL
 1 --- C.........(RSMN).....
 2 --- C.... THIS MAIN PROGRAM READS INITIAL CONDITIONS AND SETS UP THE
 3 --- C.... COORDINATE SYSTEM AND NEUTRAL ATMOSPHERE,,NOTE S(4500) FOR D300
 4 --- C-----------------------------------------------------------------
 5 --- C  S=ARRAY USED IN THE MATRIX SOLVER : D,T=ARRAYS FOR TRANSFERRING
 6 --- C  NEUTRAL DENSITIES FROM SUBR AMBS TO MAIN : FYON=A STORAGE ARRAY
 7 --- C  FOR ION PRODUCTION, USED TO PREVENT ZERO ION DENSITY AT NIGHT
 8 --- C  PROD= ARRAY FOR TRANSFERRING DATA BETWEEN SUBR PEPR AND MAIN
 9 --- C  N=ION DENSITY : TI=ION AND ELECTRON TEMPS :F=ARRAY STORING FIJ'S
10 --- C  DNDT=DN/DT FOR PREDICTION OF NEXT DENSITY : U=VELOCITY STORAGE
11 --- C  BG=VARIABLE COORDINATE FACTOR : BM=MAGNETIC FIELD STRENGTH :
12 --- C  GR=GRAVITY : R=GEOCENTRIC RADIUS TO GRID PT : SL=ARC LENGTH
13 --- C  ITEMS IN COMMON/ND/ ARE NEUTRAL DENSITIES AND TEMP ; PHION=ION
14 --- C  PRODUCTION RATE : Z=ALT ; JMAX=MAX NUMBER OF GRID PTS ; JMAX1=
15 --- C  JMAX-1 ;NSAVE,TISAV=STORAGE ARRAYS FOR PREVIOUS DENSITY/TEMP
16 --- C  FY=ION FLUX STORE ; UN=NEUTRAL WIND VELOCITY ; EHT=ELECTRON
17 --- C  HEATING RATE ; JLL AND JUL ARE LOW AND UP LIMITS FOR PRINTING
18 --- C-----------------------------------------------------------------
19 ---       IMPLICIT REAL(N)
20 ---       REAL  Z,DT,DH,THF,EPS,D,T,ZZJ,SEC,GLAT,GL,CHI,AP,DEC,ETRAN
21 ---      1     ,BLON,F107,F107A,MG,TF,FYON,SZA(300)
22 ---         INTEGER NION,NEQ
23 ---       DIMENSION S(4500),D(7),T(2),GL(300),FD(9),FYON(300),PROD(3,300)
24 ---       DIMENSION N(4,300),TI(3,300),F(2),GLAT(300),PSEC(300),DNDT(2,300)
25 ---       COMMON/VN/U(2,300),BG(300),BM(300),GR(2,300),R(2,300),SL(300)
26 ---       COMMON/ND/DN(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
27 ---       COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,NION,TF,ITF
28 ---       COMMON/SAV/NSAVE(2,300),TISAV(3,300),FY(2,300),UN(300),EHT(300)
29 ---       COMMON/LPS/EPSN,DC,IMOD,I1,I2,IPRIN,IPMX,IPP,ISKP
30 ---         COMMON/FON/JON,JLL(5),JUL(5)
31 --- C-----------------------------------------------------------------
32 --- C   ZUBDY=BDY ALT FOR DYN EQ ;;  (TF,ITF)=(1,1) FOR NORMAL RUN
33 --- C   (3,3)FOR STEADY STATE : ISW IS SWITCH FOR SYMMETRIC AMBIENTS
34 --- C   TMAX= MAX SIMULATION TIME (MINS) : DT=TIME STEP(SECS) : JPRINT
35 --- C   =SWITCH FOR PRINTING AMBIENTS AND FIELD DATA : VO=EQUATORIAL
36 --- C   VELOCITY FOR EXB DRIFT : UV=NEUTRAL WIND AMPLITUDE : ZPR=MAX
37 --- C   ALT FOR ION PROD ; JEM=MAX NO OF ENERGY STEPS IN 2-STREAM :
38 --- C   RE=RADIUS OF EARTH : AMU=ATOMIC MASS : BK=BOLTZMANN :EPSN,DC
39 --- C   ARE USED IN MODIFIED STEEPEST DESCENT IN LPS : (I1,I2)=(1,2)
40 --- C   FOR NORMAL OPERATION ,(1,1) FOR TEMP SLTNS ONLY (2,2) FOR DENSITY
41 --- C   SLTNS ONLY : IPRIN=PRINT SWITCH IN LPS: IPMX=MAX NO OF PTS
42 --- C   PRINTED : IPP=# INTERVAL BETWEEN PTS : THF=IMPLICIT-EXPLICIT
43 --- C   FRACTION : EPS=CONVERGENCE FRACTION : NION=# OF IONS : SL(1)
44 --- C   =ARC LENGTH : IDAY=YEAR+DAY : SEC=UT : BLON=MAGNETIC LONGITUDE
45 --- C   F107A,F107,AP ARE SOLAR & MAGNETIC ACTIVITY INDICES : DEC=SOLAR
46 --- C   DECLINATION : ETRAN=EPHEMERIS TRANSIT
47 --- C-----------------------------------------------------------------
48 ---         DATA ZUBDY,TF,ITF,ISW,ITAU, TMAX ,  DT  ,ILPS,JPRINT,IMOD
49 ---      >   /  3000, 3, 3 , 3 ,  0 ,  -1 ,  300  , 1 ,  22 , 1 /
50 ---         DATA VO  ,  UV  , ZPR ,JEM, RE ,    AMU    ,     BK
51 ---      >    / 0  ,   0   , 900 , 60,6370,1.6726E-24,1.3807E-16/
52 ---         DATA EPSN,DC,JSPC,I1,I2,IPRIN,IPMX,IPP,THF,EPS,NION,SL(1)
53 ---      >   /  .8  ,.5,  2 , 1, 2,  1  , 299, 1 , 1 ,1E-3,2 , 0  /
54 ---         DATA  IDAY,  SEC ,BLON,F107A,F107,AP,DEC,ETRAN,ISKP,IDNDT
```

```
55 ---      >     / 74224, 38800,-40 , 75  , 71 , 5, 18,42822, 300, 0/
56 ---        DATA JUL/22,0,0,0,0/,JLL/0,0,0,0,0/
57 --- C++++ INPUT DATA FROM FILE (1):: PCO=LSHELL ; ZO=LOWER BDY ALT ;
58 --- C++++ SCAL=SCALING FACTOR FOR COORDINATE SYSTEM
59 ---        IF(IMOD.EQ.0) READ(1) JMAX,PCO,ZO,SCAL,IA1,A2,A3,A4,A5,A6,A7,A8
60 ---      >   ,((N(I,J),J=1,JMAX),I=1,2),((TI(I,J),J=1,JMAX),I=1,3),
61 ---      >   ((DNDT(I,J),J=1,JMAX),I=1,2)
62 ---        IF(IMOD.EQ.1) READ(1) JMAX,PCO,ZO,SCAL,IDAY,SEC,ETRAN,F107
63 ---      >   ,F107A,AP,BLON,DEC
64 ---      >   ,((N(I,J),J=1,JMAX),I=1,2),((TI(I,J),J=1,JMAX),I=1,3),
65 ---      >   ((DNDT(I,J),J=1,JMAX),I=1,2)
66 ---           JMAX1=JMAX-1
67 ---           NEQ=2*JMAX-4
68 ---           J2MAX=2*JMAX-1
69 ---           IEQM1=JMAX/2
70 ---           IF(JPRINT.GT.JMAX) JPRINT=JMAX
71 ---        WRITE(6,114)
72 ---  114   FORMAT(/8X,*IDAY*,6X,*MCL*,6X,*SCAL*,6X,*ZO*,6X,*ZUBDY*,6X,
73 ---      >   *DEC*,6X,*ETRAN*,6X,*BLON*,6X,*TMAX*,6X,*F107*)
74 ---        WRITE(6,103) IDAY,PCO,SCAL,ZO,ZUBDY,DEC,ETRAN,BLON,TMAX,F107
75 ---        WRITE(6,115)
76 ---  115   FORMAT(/9X,*JEM*,7X,*AP*,7X,*ZPR*,7X,*UV*,7X,*VO*,7X,*F107A*)
77 ---        WRITE(6,103) JEM,AP,ZPR,UV,VO,F107A,SAT,N(2,150),TI(3,150)
78 --- C***TIME  STEP : END NEAR LINE 195
79 ---        DO 221 JTI=1,100
80 --- C-----
81 ---         WRITE(6,117)
82 ---  117    FORMAT(/9X,*ITAU*,7X,*DT*,7X,*DT1*,7X,*UT*,7X,*SZA*,7X,
83 ---      >    *SZAC*,7X,*SAT*)
84 ---         WRITE(6,103) ITAU,DT,DT1,SEC,SZA(JCHI),SZA(JCHIC),SAT
85 ---           WRITE(2) JMAX,PCO,ZO,SCAL,IDAY,SEC,ETRAN,F107,F107A,AP,BLON
86 ---      1 ,DEC,((N(I,J),J=1,JMAX),I=1,2),((TI(I,J),J=1,JMAX),I=1,3),
87 ---      2 ((DNDT(I,J),J=1,JMAX),I=1,2)
88 ---           REWIND 2
89 ---        IF(ITAU-DT.GT.60*TMAX) STOP
90 --- C-----
91 --- C.... IF NO EXB DRIFT(VO=0) DON'T RECALCULATE FIELD PARAMATERS....
92 ---           IVO=0
93 ---         IF(VO.EQ.0)  IVO=0
94 ---         IF(JTI.EQ.1) IVO=1
95 ---         IF(IVO.EQ.0)  GO TO 1000
96 ---          RPTS=(JMAX+1)/2-1
97 ---          DH=1/RPTS
98 --- C... THIS LOOP DETERMINES FIELD PARAMETERS AT BOTH FULL AND HALF
99 --- C... POINTS ON ODD/EVEN IJ ; AND FOR BOTH HEMISPHERES
100 ---        DO 950 IJ=1,J2MAX
101 ---          CALL FIELD(IJ,J2MAX,PCO,RE,ZO,SCAL,X,FD)
102 ---          J=(IJ+1)/2
103 ---          JU=JMAX+1-J
104 ---          IF(IJ.EQ.1) SREF=FD(8)*1.E5
105 ---          IF((IJ/2)*2.EQ.IJ) GO TO 975
106 ---          Z(J)=FD(1)
107 ---          IF(Z(J).LT.ZUBDY) JB=J
108 ---          IF(Z(J).LT.400) JCHI=J
109 ---          GL(J)=FD(4)
```

```
10 ---           BM(J)=FD(6)
11 ---           BG(J)=FD(7)
12 ---           UN(J)=UV*FD(3)
13 ---           GR(2,J)=-FD(5)
14 ---           R(2,J)=(RE+FD(1))*1.E5
15 ---           IF(J.NE.1) SL(J)=SREF-FD(8)*1.E5
16 ---           IF(GL(J).EQ.0) GO TO 1000
17 ---           GL(JU)=-GL(J)
18 ---           Z(JU)=Z(J)
19 ---           BM(JU)=BM(J)
20 ---           BG(JU)=BG(J)
21 ---           UN(JU)=-UN(J)
22 ---           GR(2,JU)=-GR(2,J)
23 ---           R(2,JU)=R(2,J)
24 ---          SL(JU)=SREF+FD(8)*1.E5
25 ---           GO TO 950
26 --- 975          CONTINUE
27 ---           GR(1,J+1)=-FD(5)
28 ---           R(1,J)=(RE+FD(1))*1.E5
29 ---           GR(1,JU)=-GR(1,J+1)
30 ---           R(1,JU-1)=R(1,J)
31 --- 950          CONTINUE
32 --- 1000         CONTINUE
33 --- C++++++ CALL CMINOR TO GET N(NO+) AND N(O2+) ;; CALL AMBS TO GET NEUTRALS
34 ---           DO 17 I=1,JMAX
35 ---         IF(JTI.EQ.1) U(1,I)=0
36 ---         IF(JTI.EQ.1) U(2,I)=0
37 ---         IF(JTI.EQ.1) FY(1,I)=0.
38 ---         IF(JTI.EQ.1) FY(2,I)=0.
39 ---         IF(IDNDT+JTI.EQ.1) DNDT(1,I)=0.0
40 ---         IF(IDNDT+JTI.EQ.1) DNDT(2,I)=0.0
41 ---           CALL CMINOR(I,N,TI,ENOP,ENO2P)
42 ---           N(3,I)=ENOP+ENO2P
43 ---            ZZJ=Z(I)
44 ---            GLTM=GL(I)
45 ---           CALL AMBS(I,IDAY,SEC,ZZJ,GLTM,DEC,ETRAN,F107,F107A,AP,BLON
46 ---        > ,D,T,CHI,IY,IDY,SAT,ID,GLATD)
47 ---         ON(I)=D(2)
48 ---         HN(I)=D(7)
49 ---         N2N(I)=D(3)
50 ---         O2N(I)=0.5*D(4)
51 ---         TN(I)=T(2)
52 ---         SZA(I)=CHI
53 ---         GLAT(I)=GLATD
54 ---         PHION(I)=0.0
55 ---         EHT(I)=0.0
56 --- 17           CONTINUE
57 --- C... JCHIC=CONJUGATE OF JCHI ; THE NEXT FEW LINES SETS THE NEW
58 --- C... TIME STEP ACCORDING TO THE DIFFICULTY OF THE PREVIOUS SOLTN
59 --- C... AND THE SOLAR ZENITH ANGLE IN BOTH HEMISPHERES
60 ---           JCHIC=JMAX+1-JCHI
61 ---           IF(ITER.LE.7) DT1=+300
62 ---           IF(ITER.GT.7) DT1=-300
63 ---           IF((SZA(JCHI).GT.1.4).AND.(SZA(JCHI).LT.2.3)) DT1=-300
64 ---           IF((SZA(JCHIC).GT.1.4).AND.(SZA(JCHIC).LT.2.3)) DT1=-300
```

```
65 ---          DT=DT+DT1
66 ---          IF(DT.GT.1800) DT=1800
67 ---          IF(DT.LT.300) DT=300
68 ---          IF(JTI.LE.9) DT=300
69 --- C... IF THE SUN ISN'T SHINING IN EITHER HEMISPHERE DON'T CALC PROD
70 ---          IF((SZA(JCHI).GT.1.57).AND.(SZA(JCHIC).GT.1.57)) GO TO 23
71 --- C... PEPR=PRIMARY PE PROD PROG; PE2S=2-STREAM PROG TO GET ELEC
72 --- C... HEATING RATE; NOTE THAT FYON IS USED TO PREVENT PHION FROM
73 --- C... GOING TO ZERO;; PRIMARY SPECT STORE TEMPORARILY ON DCB 8
74 ---         REWIND 8
75 ---         CALL PEPR(Z,ON,N2N,O2N,SZA,TN,ZPR,JMAX,O,PROD,JEM)
76 ---         REWIND 8
77 ---         CALL PE2S(Z,ON,N2N,O2N,BM,BG,JMAX,DH,N,TI,EHT,PSEC)
78 --- C**** PROD(1,I=1-3) = 4S,2D,2P ION STATES OF O+ ; O4DL=LOSS OF O+(2D)
79 --- C**** TO O+(4S) AND ON2PL=LOSS OF O+(2D) TO N2+. THE RATIO IS USED TO
80 --- C**** CALC FRACTION OF O+(2D) ENDING UP AS O+(4S) : SEE TORR+TORR REV
81 --- C**** GEOPHYS AUG 1978 P330
82 ---          DO 20 I=1,JMAX
83 ---          IF(JTI.EQ.1) FYON(I)=.001*PROD(1,I)
84 ---          O4DL=2.0E-11*ON(I)+7.8E-8*SQRT(300/TI(3,I))*N(1,I)+4.0E-11
85 ---        >    *N2N(I)
86 ---          ON2PL=1.0E-10*N2N(I)
87 ---          OMETAS=PROD(2,I)+PROD(3,I)
88 ---          PHION(I)=PROD(1,I)+OMETAS*O4DL/(O4DL+ON2PL)+FYON(I)+PSEC(I)
89 ---   20      CONTINUE
90 ---   23      CONTINUE
91 --- C..... GENERATE SYMMETRIC AMBIENTS IF DESIRED .....
92 ---          IF(ISW.LT.3) GO TO 25
93 ---          DO 24 I=1,IEQM1
94 ---          JU=JMAX+1-I
95 ---          ON(JU)=ON(I)
96 ---          N2N(JU)=N2N(I)
97 ---          O2N(JU)=O2N(I)
98 ---          HN(JU)=HN(I)
99 ---          TN(JU)=TN(I)
00 ---          PHION(JU)=PHION(I)
01 ---          EHT(JU)=EHT(I)
02 ---          DO 13 J=1,3
03 ---          N(J,JU)=N(J,I)
04 ---   13     TI(J,JU)=TI(J,I)
05 ---   24     CONTINUE
06 ---   25     CONTINUE
07 ---         IF(ILPS.NE.0) CALL LOOPS(S,NEQ,N,TI,ITAU,JB,SEC,DNDT)
08 --- C.... PRINTING OF AMBIENT PARAMATERS.................
09 ---          IF((JPRINT.EQ.0).OR.(ITF.NE.3)) GO TO 119
10 ---          DO 106 J=1,JPRINT,1
11 --- C...TEMPORARY CALC OF MINOR IONS
12 ---          CALL CMINOR(J,N,TI,ENOP,ENO2P)
13 ---          IF((J/30)*30.EQ.J) WRITE(6,112)
14 ---          IF(J.EQ.1) WRITE(6,112)
15 ---         IF((J/JSPC)*JSPC.NE.J) GO TO 106
16 ---   112    FORMAT(/2X,*J*,6X,*ALT*,5X,*GL*,5X,*BM*,5X,*SZA*,5X,*GR*,4X
17 ---       >  ,*TN*,5X,*N(O)*,6X,*N(H)*,6X,*N(N2)*,6X,*N(O2)*,6X,*PROD*
18 ---       >  ,6X,*HEAT*,6X,*BG*,4X,*GLAT*)
19 ---          WRITE(6,111) J,Z(J),GL(J),BM(J),SZA(J),GR(1,J),TN(J),ON(J)
```

```
20 ---      >       ,HN(J),N2N(J),O2N(J),PHION(J),EHT(J),ENOP,ENO2P
21 ---  106      CONTINUE
22 --- 119      CONTINUE
23 --- C... IF PREVIOUS TIME STEP IS SUCCESSFUL ADVANCE TIME
24 ---         IF(ITF.EQ.4) GO TO 23
25 ---         IF(ITF.EQ.1) SEC=SEC+THF*DT
26 ---         ITAU=ITAU+DT
27 ---         IF((I1+I2.NE.3).AND.(ITAU-DT.GT.TMAX*60)) STOP
28 ---         IF(I1+I2.NE.3) GO TO 221
29 ---  221      CONTINUE
30 ---  102  FORMAT(1H ,I10,1P11E11.2)
31 ---  103    FORMAT(1H ,I10,11F10.2)
32 ---  111    FORMAT(I4,F8.0,3F7.2,2F7.0,1P9E10.1)
33 ---    58   FORMAT(8(1PE10.3))
34 ---   330 FORMAT(3I10)
35 ---         END
36 --- C..............<RSFO>.................
37 ---         SUBROUTINE FIELD(J,JMAX,PCO,RE,ZO,SCAL,X,FD)
38 --- C-------------------------------------------------
39 --- C  THIS PROGRAM DETERMINES THE GRID POINT SPACING GIVEN JUST JMAX=
40 --- C  # OF GRID POINTS, PCO=LSHELL, ZO=LOWER BOUNDARY, SCAL=SCALING
41 --- C  FACTOR;; X=COLATITUDE, THE FIELD PARAMETERS ARE TRANSFERRED
42 --- C  THROUGH FD(I);; RO=EQUATORIAL RADIUS TO FLUX TUBE, INITIAL VALUES
43 --- C  FOR X AND R ARE SET. DH=DISTANCE BETWEEN POINTS IN THE X COORDINATE
44 --- C-------------------------------------------------
45 ---         IMPLICIT REAL(N)
46 ---           DIMENSION FD(9)
47 ---           IF(J.NE.1) GO TO 950
48 ---            RO=RE*PCO
49 ---           R=RO
50 ---           X=.5
51 ---           PTS=(JMAX+1)/2
52 ---           IPTS=PTS
53 --- C'''' SCALK IS A SCALING FACTOR TO ENSURE THAT THE X-COORD.
54 --- C'''' RANGES FROM 0 TO 1
55 ---          RAT=(RE+ZO)/RE
56 ---         QMAX=(SQRT(1-RAT/PCO))/(RAT**2)
57 ---         SCALK=1/SINH(SCAL*(QMAX))
58 ---           DH=1/(PTS-1)
59 ---           RAT=(RE/RO)**2
60 --- C-----ESTABLISH PTS. ON FIELD LINE
61 --- C.... R=RADIUS TO FIELD PT ; FD1=ALT ; FD2=SIN(DIP ANG) ; FD4=GEOM LAT
62 --- C.... FD5=GRAVITY ; FD6 IS PROP TO MAG FIELD STRENGTH ; FD7=VARIABLE
63 --- C.... COORD FACTOR ; DS=STEP SIZE(KM) ; FD8=ARC LENGTH FROM EQUAT (KM
64 --- C
65 --- 950       DX=1-(J-1)/(PTS-1)
66 ---           IF(J.EQ.IPTS)     GO TO 4
67 --- C... Q=THE DIPOLE COORD  DETERMINED FROM -- SINH(KQ)=DX
68 ---            SCX=DX/SCALK
69 ---         Q=ALOG(SCX+SQRT(SCX**2+1))/SCAL
70 ---     3   SHX=SIN(X)
71 ---         CHX=COS(X)
72 --- C... THE NEXT 6 LINES ARE A NEWTON SOLVER FOR THE EQUATION F(X)=0
73 --- C... THIS DETERMINES THE COLATITUDE X
74 ---           FEX=(RO**2)*(SHX**4)-(RE**2)*CHX/Q
```

```
75 ---           DEX=(RO**2)*4*(SHX**3)*CHX+(RE**2)*SHX/Q
76 ---           X=X-FEX/DEX
77 ---           IF(ABS(FEX/(DEX*X)).GT.1.0E-6) GO TO 3
78 ---           GO TO 5
79 --- 4         X=1.570796327
80 ---           SHX=1
81 ---           CHX=0
82 --- C ----LAT. RADIUS. DIP
83 --- 5         R=RO*SHX**2
84 ---           FD(1)=R-RE
85 ---         SQTH=SQRT(3*(CHX**2)+1)
86 ---           FD(2)=2*CHX/SQTH
87 ---           FD(3)=SHX/SQTH
88 ---           FD(4)=1.570796327-X
89 ---           FD(5)=FD(2)*3.98E+10/((RE+FD(1))**2)
90 ---          FD(6)=8.271E+25*SQTH/(R*1.E+5)**3
91 ---         FD(7)=SCAL*COSH(SCAL*Q)*SCALK*1.E-5*RE**2*SQTH/R**3
92 ---           DS=1.0E-5*DH/FD(7)
93 ---         XX=ALOG(1.732*CHX+SQTH)
94 ---         FD(8)=RO*.23868*(XX+SINH(XX)*COSH(XX))
95 ---           RETURN
96 ---           END
97 --- C.....................................
98 ---           SUBROUTINE AMBS(J,IDAY,SEC,ZZJ,GL,DEC,ETRAN,F107,F107A,AP,
99 ---       >   BLON,D,T,SZA,NY,NDY,SAT,ID,GLATD)
00 --- C-------------------------------------------------------
01 --- C  THIS PROGRAM EVALUATES THE SOLAR ZENITH ANGLE, DETERMINES THE
02 --- C  THE DAY AND TIME AND CALLS A.E. HEDINS MSIS MODEL(GTS3S) TO GET
03 --- C  THE NEUTRAL DENSITIES AND TEMPERATURE
04 --- C-------------------------------------------------------
05 ---           DATA  PLAT , PLON ,  PI  ,ICNT
06 ---       >   /   1.375 , 1.222 , 3.14159, 0 /
07 --- C---------CORRECT TIME FOR FULL DAYS AND YEARS---
08 --- 999      IF(J.NE.1) GO TO 1000
09 ---           DECL=DEC*3.14159/180
10 ---         NY=IDAY/1000
11 ---         NDY=MOD(IDAY,1000)
12 ---         IF(MOD(NY,4).EQ.0)LY=366
13 ---         IF(MOD(NY,4).NE.0)LY=365
14 ---         SX=AMOD(SEC,86400.)
15 ---         NDY=NDY+INT(SEC/86400)
16 ---         NY=NY+NDY/LY
17 ---         ID=1000*NY+MOD(NDY,LY)
18 --- 1000        CONTINUE
19 --- C---------TRANSFORM MAGNETIC TO GEOGRAPHIC COORDINATES---
20 --- C-------- AND CALCULATE SZ  ANGLE AT EACH FIELD LINE POINT---
21 ---         BLOR=BLON*PI/180.0
22 ---         XM=COS(GL)*COS(BLOR-PLON)
23 ---         YM=COS(GL)*SIN(BLOR-PLON)
24 ---         ZM=SIN(GL)
25 ---         XG=XM*SIN(PLAT)+ZM*COS(PLAT)
26 ---         YG=YM
27 ---         ZG=-XM*COS(PLAT)+ZM*SIN(PLAT)
28 ---         GLAT=ASIN(ZG)
29 ---         GLATD=GLAT*180./PI
```

```
30 ---        GLON=(PLON+ATAN2(YG,XG))*180/PI
31 ---        SAT=(SX-ETRAN+43200.0)/3600-GLON/15.0
32 ---        HH=(SAT-12.)*15*PI/180.
33 ---        SZA=ACOS(COS(GLAT)*COS(DECL)*COS(HH)
34 ---       >  +SIN(GLAT)*SIN(DECL))
35 --- C...   CALL GTS3S FOR NEUTRAL PARAMS.......
36 ---        CALL GTS3S(ID,SX,ZZJ,GLATD,GLON,SAT,F107A,F107,AP,48,D,T)
37 ---          RETURN
38 ---          END

ION DATASET IS ABOUT TO BE WRITTEN

VEN,  RF DONE
ION DATASET HAS NOW BEEN WRITTEN

-+-+-+-+-+-+-+    EDITOR TERMINATING                    80/100 SECONDS IS ELAPSED TIM
```

```
IT,D=PLIB,DN=RS2S,DL
 1 --- C.............<RS2S>.....................
 2 ---           SUBROUTINE  PE2S(Z,ZOX,ZN2,ZO2,BM,BG,IMAX,DH,N,TI,EHEAT,PSEC)
 3 --- C++++ THE ORIGINAL PROGRAM WAS WRITTEN BY A. NAGY AND P. BANKS. THIS
 4 --- C++++ VERSION WAS IMPROVED BY P. RICHARDS TO COVER ENTIRE FIELD LINE
 5 --- C++++ AND TO TAKE INTO ACCOUNT PITCH ANGLE TRAPPING IN AN AD HOC WAY
 6 --- C++++ PROGRAM FOR SOLVING THE 2-STREAM EQUATIONS FOR PHOTOELECTRON
 7 --- C++++ FLUXES AS GIVEN IN BANKS AND KOCKARTS 1973 ;THE CALCULATIONS BEGIN
 8 --- C++++ AT 100 EV AND WORK DOWN IN 1 EV STEPS - SECONDARY AND DEGRADED
 9 --- C++++ PRIMARIES  BEING INCLUDED AT LOWER ENERGIES
10 --- C---- VARIOUS PRINTS CAN BE OBTAINED BY USING ISW=2 -PRINT # OF PTS WITH
11 --- C---- EXCESSIVE HEAT LOSS :: ISW=3 PRINT COEFFS OF FLUX DE'S :: ISW=4
12 --- C---- PRINT FLUXES AT EACH ENERGY :: ISW=5 PRINT TOTAL FLUXES, HEATING RATES
13 ---        REAL N2SIGA,N2SIGS,N2PE,N2PI,N2IONS,N
14 ---        DIMENSION N2SIGA(100),O2SIGA(100),OXSIGA(100),TPROD(300),
15 ---       7    TSIGNE(300),O1DSIG(100),O1SSIG(100),REDEXC(300),REDEM(300),
16 ---       8    SECION(300),N2PE(100),O2PE(100),OXPE(100),EHEAT(300),E(100),
17 ---       8   SUMUP(300),SUMDWN(300),SN2ION(300),PSEC(300),PHIS(2,300)
18 ---         DIMENSION XNU(3,10),OMEG(3,10),P(3,10),W(3,10),AD(3,10),
19 ---       1     GAM(3,10),SIGEL(3,100),NN(3)
20 ---          DIMENSION Z(300),ZOX(300),ZN2(300),ZO2(300),BM(300),BG(300),
21 ---       1    N(4,300),TI(3,300)
22 ---         COMMON/TRI/PROD(300),PRODWN(300,100),PRODUP(300,100),
23 ---       1     PHIDWN(300),PHIUP(300),T1(300),T2(300),DS(300)
24 ---          COMMON/PANGS/ZPAS,PASK,IPAS,IPASC,FPAS
25 --- C.... ISW=PRINT SWITCH, JEP=ENERGY PRINT SWITCH, IPRIN1=PRINT SWITCH
26 --- C.... JMIN=LOWEST ENERGY PE, M1=FIRST PT ON FIELD LINE, AVMU=<COSO>
27 --- C.... VTOT=VOLUME FLUX TUBE, FAC=CONJ ILLUM FACTOR, FPAS=PITCH ANGLE
28 --- C.... SCATT FACTOR, JMAX=HIGHEST ENERGY PE, KITER=# OF ALLOWED ITERS
29 --- C.... M=BOUNDARY OF PDE SOLUTION, ZPRIN=PRINT BDY, ZPAS=ALT OF PAS
30 ---          DATA  ISW , JEP , IPRIN1 , IJEP , JMIN , M1 ,AVMU , VTOT,FAC2
31 ---       1    /    2  ,  10 ,   10   , 1  ,  1   ,  1 , .577  , 0  , 1/
32 ---          DATA   FAC , FPAS , JSN , JMAX, M ,ZPRIN ,ZPAS,EFLUX,JTI
33 ---       1       / 1  ,   .8 ,  3  , 100 , 41, 1.E9 ,1000 ,  0 , 0 /
34 ---         READ(8,330) IPMAX,JMAX
35 ---          IPMAC=IMAX+1-IPMAX
36 ---          IF(Z(M).GT.ZPAS) ZPAS=Z(M)
37 ---          IEQ=(IMAX+1)/2
38 ---         IMAX1=IMAX-1
39 ---         IPMAX1=IPMAX+1
40 ---         IEQP=IEQ-IPMAX
41 ---         IPMAX2=IPMAX*2
42 ---         M2=M1+1
43 ---         ITEST=0
44 ---         VTOT=0.0
45 ---         IMXM=IMAX+1-M
46 ---         JTI=JTI+1
47 ---         REWIND 7
48 ---         IDSC=7
49 ---         IF(JTI.EQ.1) IDSC=5
50 ---         IF(JTI.NE.1) GO TO 115
51 --- C.. IMAX=MAX ALT. INCS; IPMAX=MAX ALT. INCS FOR PROD.
52 --- C**** PARAMETERS FOR PROD OF SECONDARIES; BANKS ET AL JGR 1974 P1459
53 --- C**** SIGEL = ELASTIC SCATT X-SECT AT ENERGY J
54 ---          READ(5,330) NN(1),NN(2),NN(3)
```

```
      DO 101 I=1,3
     NNI=NN(I)
     DO 101 J=1,NNI
101     READ(5,305)   XNU(I,J),OMEG(I,J),P(I,J),W(I,J)
    1    ,AD(I,J),GAM(I,J)
     DO 102 I=1,3
102     READ(5,304) (SIGEL(I,J),J=1,12)
C,,,, AD(1,1) IS A G-STOL PARAM ; N2PI IS INEL. B-S FRACTION
C,,,, FACTP =FACTOR FOR EXTRAPOLATING PROD RATES
     AD(1,1)=1.5E5
       DATA  N2PI , O2PI , OXPI , FACTP ,   QQ
    1      /  0.5 ,  0.5 , 0.5  ,  .7   , 6.51E-14  /
C>>>> EXTRAPOLATION OF X-SECTION PARAMS (TURR-WEDDE ???)
     DO 2 JE=13,JMAX
      DO 2 I=1,3
     EJ=JE
     EJ=EJ-0.5
     EFACT=(1.0-(SQRT(EJ)-3.464)/20.)
2    SIGEL(I,JE)=SIGEL(I,12)*EFACT
CEEEE ELASTIC B-S PROBABILITIES -- N2PE ETC. EEEEE
     DO 3 MK=1,12
     N2PE(MK)=0.5
     O2PE(MK)=0.5
   3 OXPE(MK)=0.5
     DO 4 MK=13,JMAX
     EK=MK
     EK=EK-0.5
     EFAC=SQRT(12./EK)
     N2PE(MK)=0.5*EFAC
     O2PE(MK)=0.5*EFAC
   4 OXPE(MK)=0.5*EFAC
C+++++ HIGHEST ENERGY SET FOR J ; O(1S),O(1D) X-S EVALUATED
115     J=100
     DO 7 JC=1,10
     O1SSIG(JC)=0.0
   7 O1DSIG(JC)=0.0
     DO 15 MM=3,J
     EJ=MM
     EJ=EJ-0.5
     IF(MM.LT.5) GO TO 15
     O1SSIG(MM)=((6.51E-14)*(4.2E-3)/(4.17*4.17))*(1.-(4.17/EJ)**0.5)
    1   *(4.17/EJ)
  15 O1DSIG(MM)=((6.51E-14)*(1.0E-2)/(1.96*1.96))*(1.-1.96/EJ)**2
    1   *1.96/EJ
C---- SN2ION= TOT. SEC. IONS FROM N2  ; EHEAT=ELECTRON HEATING RATE ; TPROD=
C----PHINET= TOTAL  PE FLUX ; SUMUP=TOTAL  UP FLUX ;SECION=TOTAL  SECONDARY
C---- IONIZATION ; REDEXC=EXCITATION OF 6300 LINE
C.... ARC LENGTH DS(I) AND TOTAL FLUX TUBE VOL (VTOT) ARE OBTAINED
C.... IPAS=GRID POINT FOR TRANSITION TO PITCH ANGLE TRAPPING
     DO 16 I=M1,IMAX
     IF(I.GT.1) DS(I)=.5*(DH/BG(I)+DH/BG(I-1))
       IF(Z(I).GT.ZPAS) VTOT=VTOT+DS(I)/(BM(I)*2.038E+8)
       IF((Z(I).LT.ZPAS).AND.(I.LT.IEQ)) IPAS=I
     TSIGNE(I)=0
     PHIDWN(I)=0
```

```
      PHIUP(I)=0
      SN2ION(I)=0.0
      EHEAT(I)=0.0
      TPROD(I)=0.0
      SUMUP(I)=0.
      SUMDWN(I)=0.
      SECION(I)=0.0
      REDEXC(I)=0.0
      PSEC(I)=0.0
   16  CONTINUE
C#### SET ENERGY STEPS ;; PRODUP=Q+ , PRODWN =Q- AND PROD=Q ; BEK 258
C#### IPASC=CONJUGATE GRID POINT FOR PITCH ANGLE TRAPPING, PASK=A
C#### CONSTANT ASSOCIATED WITH PAS CALCULATION
        IPASC=2*IEQ-IPAS
        PASK=FPAS/(VTOT*2.038E+8*BM(IPAS))
      E(1)=0.5
      DO 92 JK=2,JMAX
   92 E(JK)=E(JK-1)+1.
      DO 22 I=M1,IMAX
      DO 21 JJ=1,JMAX
      PRODUP(I,JJ)=1.0E-20
   21 PRODWN(I,JJ)=1.0E-20
   22 CONTINUE
C\\\\\\\\\\\\\\
C//////////////MAIN CALCULATIONS  BEGIN HERE ////////////
   23 DO 6 I=M1,IMAX
    6 PROD(I)=1.0E-20
      IF(ISW.EQ.3) WRITE(6,330) J
      IF(J.GT.JMAX) GO TO 98
C@@@@ PROD IS READ FROM A FILE CREATED BY PEPROD THEN
C@@@@ EXTRAPOLATED TO HIGHER ALTS. IF NECESSARY
    8 READ(8,336) (PROD(I), I=1,IPMAX2)
        DO 117 I=1,IPMAX
  117   PROD(IPMAC-1+I)=PROD(IPMAX+I)
      DO 119 I=IPMAX1,IEQ
  119 PROD(I)=PROD(I-1)*FACTP
      DO 19 I=1,IEQIP
   19 PROD(IPMAC-I)=PROD(IPMAC+1-I)*FACTP
      DO 96 I=M1,IMAX
      PROD(I)=PROD(I)/AVMU
   96   IF(I.GT.IEQ) PROD(I)=PROD(I)*FAC
C.... TN2SA= SUM OF N2SIGA(TOT INEL. X-SECT BEK 258) 10 LINES DOWN
C.... THESE ARE EVALUATED  IN  UCSIGMA AND STORED ON UCSRESULTS
C.... NCAR VERSION READS FROM 5 AT FIRST THEN READS AND  WRITES 7
   98 READ(IDSC,330) JPRINT
      READ(IDSC,335) TN2SA,TO2SA,TOXSA
      READ(IDSC,336) (N2SIGA(K), K=1,JPRINT)
      READ(IDSC,336) (O2SIGA(K), K=1,JPRINT)
      READ(IDSC,336) (OXSIGA(K), K=1,JPRINT)
      IF(JTI.NE.1) GO TO 198
      WRITE(7,330) JPRINT
      WRITE(7,335) TN2SA,TO2SA,TOXSA
      WRITE(7,336) (N2SIGA(K), K=1,JPRINT)
      WRITE(7,336) (O2SIGA(K), K=1,JPRINT)
      WRITE(7,336) (OXSIGA(K), K=1,JPRINT)
```

```
      198   JPRIN1=JPRINT+1
            DO 1861 K=JPRIN1, J
            N2SIGA(K)=0.0
            O2SIGA(K)=0.0
     1861   OXSIGA(K)=0.0
C&&&& ELECTRON-ELECTRON CONTINUOUS ENERGY LOSS
C!!!! SWARTZ ET AL JGR 1971 P8425--- THE FOLL. SWITCH
C---- ALLOWS A START FROM LOWER EMAX WITH SAME INPUT FILE 7
              IF(J.GT.JMAX)   GO TO 82
              DE=0.0
            DO 11 I=M2,IMAX1
              ET=8.618E-5*T1(3,I)
              ZNE=N(1,I)+N(2,I)+N(3,I)
            TSIGNE(I)=((3.37E-12*ZNE**0.97)/(E(J)**0.94))*((E(J)-ET)/
           1   (E(J)-(0.53*ET)))**2.36
            TSIGNE(I)=TSIGNE(I)/AVMU
              DELZ=DS(I)
              IF(I.GT.IEQ) DELZ=DS(I+1)
            TEST=TSIGNE(I)*DELZ
              IF(Z(I).GT.1000) DE=DE+TEST
            IF (TEST .LT. 1.0) GO TO 11
            IF(J.EQ.2) ITEST=ITEST+1
C           IF(J.GT.1) WRITE(6,311) I,Z(I),TEST
            TSIGNE(I)=1.0/DELZ
         11 CONTINUE
C           WRITE(6,26)
C           WRITE(6,27) TN2SA,TO2SA,TOXSA,SIGEL(1,J),SIGEL(2,J),SIGEL(3,J)
C          1     ,O1DSIG(J),O1SSIG(J)
C---- T1,T2 ARE COEEFS OF D.E. B&K P258 .............
            IF(ISW.EQ.3) WRITE(6,339)
      339   FORMAT(5X,*I*,7X,*ALT*,8X,*T1*,8X,*T2*,7X,*TSIGNE*,7X,*PROD*)
            DO 30 I=M1,IMAX
            T1(I)=ZN2(I)*SIGEL(1,J)*N2PE(J)+ZO2(I)*SIGEL(2,J)*O2PE(J)+ZOX(I)
           2     *SIGEL(3,J)*OXPE(J)
            T1(I)=T1(I)/AVMU
            T2(I)=ZN2(I)*(SIGEL(1,J)*N2PE(J)+TN2SA)+ZO2(I)*(SIGEL(2,J)*
           2     O2PE(J)+TO2SA)+ZOX(I)*(SIGEL(3,J)*OXPE(J)+TOXSA)
            T2(I)=(T2(I)/AVMU+TSIGNE(I))
            IF(ISW.EQ.3)WRITE(6,311) I,Z(I),T1(I),T2(I),TSIGNE(I),PROD(I)
           >    ,N(1,I),N(2,I),T1(3,I),BM(I),BG(I),DS(I)
         30 CONTINUE
C---- SET BOUNDARY CONDITIONS ON FLUXES ....
              IF(J.EQ.1) GO TO 56
              PHIDWN(M1)=.5*PROD(M1)/(T2(M1)-T1(M1))/AVMU
              PHIDWN(IMAX)=.5*PROD(IMAX)/(T2(IMAX)-T1(IMAX))/AVMU
              PHIUP(M1)=PHIDWN(M1)
              PHIUP(IMAX)=PHIDWN(IMAX)
C,,,, DO LOOP FOR ITERATING SOLNS TO CONVERGENCE
      56      DO 161 KIT=1,50
              DO 58 IJ=M2,IMAX1
              PHIS(1,IJ)=PHIDWN(IJ)
      58      PHIS(2,IJ)=PHIUP(IJ)
               CALL TRIS(1,M2,M,J,M,BM,BG,Z,IMAX,DH)
               CALL TRIS(-1,IMXM,IMAX1,J,M,BM,BG,Z,IMAX,DH)
C++++ TESTING FOR CONVERGENCE TO A SOLUTION ++++
```

```
220 ---            IDIV=0
221 ---           DO 61 I=M2,IMAX1
222 ---           IF(ABS((PHIUP(I)-PHIS(2,I))/PHIUP(I)).GT.1.E-4) IDIV=IDIV+1
223 --- 61        IF(ABS((PHIDWN(I)-PHIS(1,I))/PHIDWN(I)).GT.1.E-4) IDIV=IDIV+1
224 ---           IF(IDIV.EQ.0) GO TO 163
225 --- 161          CONTINUE
226 --- 63          CONTINUE
227 --- C----
228 --- C---- SUM FLUXES AND PRINT RESULTS ;; FYUP & FYDWN ARE USED ONLY TO PRINT
229 --- 163       DO 118 I=M1,IMAX
230 ---         SUMUP(I)=SUMUP(I)+PHIUP(I)*AVMU
231 ---         SUMDWN(I)=SUMDWN(I)+PHIDWN(I)*AVMU
232 ---           FYUP=PHIUP(I)*AVMU
233 ---           FYDWN=PHIDWN(I)*AVMU
234 ---          IF(ISW.NE.4) GO TO 118
235 --- C....   PRINT SPECIFICATIONS ....
236 ---           IPRIN1=2
237 ---           IF(Z(I).GT.ZPRIN) IPRIN1=4
238 ---           IF(IJEP-1) 165,164,118
239 --- 164       IF(JEP.LT.15) IPRIN1=10
240 ---           IF(IABS(I-IEQ).LT.3) IPRIN1=1
241 ---           IF(J.LT.60) JEP=1
242 ---           IF(J.GE.60) JEP=10
243 --- 165       IF(J.GE.JMAX)    GO TO 162
244 ---           IF(J.LE.JMIN+1) GO TO 162
245 ---           IF((J/JEP)*JEP.NE.J) GO TO 118
246 --- 162       IF(I.EQ.1)  WRITE(6,314) J
247 ---           IF(I.EQ.1) WRITE(6,310)
248 ---           IF(((I+1)/IPRIN1)*IPRIN1.NE.I+1) GO TO 118
249 ---           IF(ISW.EQ.4) WRITE(6,311)   I,Z(I),FYDWN,FYUP,PROD(I)
250 ---      1                ,PRODUP(I,J),PRODWN(I,J),EHEAT(I)
251 --- 118          CONTINUE
252 --- C-------
253 --- C???? THE DEGRADED PE*S ARE NOW ADDED TO THEIR CORRECT ENERGY BOX AT
254 --- C???? EACH ALTITUDE STEP
255 ---           IF (J .LT. 2) GO TO 82
256 ---           L=J-1
257 --- 99        DO 71 I=M1,IMAX
258 --- C""""  DO LOOP FOR DISCRETE LOSS OF 1,2,3,4 EV ETC. SEE D+,D- BEK P258
259 ---           DO 70 K=1,L
260 ---           LL=J-K
261 ---           PRODUA=ZN2(I)*(N2SIGA(K)*N2PI*PHIDWN(I)+(1.-N2PI)*N2SIGA(K)*PHIUP
262 ---          1(I))+ZO2(I)*(O2SIGA(K)*O2PI*PHIDWN(I)+(1.-O2PI)*O2SIGA(K)*PHIUP(I
263 ---          2))+ZOX(I)*(OXSIGA(K)*OXPI*PHIDWN(I)+(1.-OXPI)*OXSIGA(K)*PHIUP(I))
264 ---           PRODDA=ZN2(I)*(N2SIGA(K)*N2PI*PHIUP(I)+(1.-N2PI)*N2SIGA(K)*PHIDWN
265 ---          1(I))+ZO2(I)*(O2SIGA(K)*O2PI*PHIUP(I)+(1.-O2PI)*O2SIGA(K)*PHIDWN(I
266 ---          2))+ZOX(I)*(OXSIGA(K)*OXPI*PHIUP(I)+(1.-OXPI)*OXSIGA(K)*PHIDWN(I))
267 ---           PRODUP(I,LL)=PRODUP(I,LL)+PRODUA/AVMU
268 ---       70  PRODWN(I,LL)=PRODWN(I,LL)+PRODDA/AVMU
269 ---       71  CONTINUE
270 --- CEEEE ELECTRON-ELECTRON CONTINUOUS ENERGY LOSS
271 ---           DO 75 I=M1,IMAX
272 ---           PRODUP(I,L)=PRODUP(I,L)+TSIGNE(I)*PHIUP(I)
273 ---       75  PRODWN(I,L)=PRODWN(I,L)+TSIGNE(I)*PHIDWN(I)
274 --- C,,,, EHPAS=HEATING DUE TO PITCH ANGLE TRAPPING AT ENERGY J. DE=
```

```
275 --- C,,,, NORMAL LOSS IN PROTONOSPHERE  TO THERMAL ELECTRONS. ETOT=
276 --- C,,,, TOTAL LOSS FOR ALL ENERGIES. EFLUX=TOTAL ENERGY LOST BY BOTH
277 --- C,,,, PAS AND COULOMB LOSSES
278 ---           EHPAS=J*PASK*(PHIUP(IPAS)+PHIDWN(IPASC))
279 ---           IF(DE.GT.J) EHPAS=0
280 ---           IF(DE.GT.J) DE=J
281 ---           ETOT=ETOT+EHPAS
282 ---           EFLUX=EFLUX+(DE*(1-FPAS)+J*FPAS)*(PHIUP(IPAS)+PHIDWN(IPASC))
283 ---         1    *AVMU
284 --- C.... EHAV=THE AVERAGE HEATING RATE IN THE FLUX TUBE. EHPAS IS ADDED
285 --- C,,,, TO COULOMB HEATING RATE
286 ---         DO 76 I=M1,IMAX
287 ---         TPROD(I)=TPROD(I)+PROD(I)
288 ---         EHEAT(I)=EHEAT(I)+TSIGNE(I)*(PHIUP(I)+PHIDWN(I))*FAC2
289 ---           IF(EHEAT(I).LT.0) WRITE(6,311) I,Z(I),PROD(I),PHIUP(I)
290 ---          IF(Z(I).GT.ZPAS) EHAV=EHAV+FAC2*
291 ---         1 TSIGNE(I)*(PHIUP(I)+PHIDWN(I))*DS(I)/(2.038E+8*BM(I))/VTOT
292 ---           IF(Z(I).GT.ZPAS) EHEAT(I)=EHEAT(I)+EHPAS
293 ---          REDEXC(I)=REDEXC(I)+(PHIUP(I)+PHIDWN(I))*ZDX(I)
294 ---         >    *(O1DSIG(J)+0.952*O1SSIG(J))
295 --- 76        CONTINUE
296 --- C*****
297 --- C***** PROD. OF SECONDARY ELECTRONS FOR E>17 EV ****
298 ---         IF (J-17) 82,73,73
299 ---      73 MAX=(J+17)/2
300 ---         TN2ION=0.0
301 ---         TO2ION=0.0
302 ---         TOXION=0.0
303 --- C---- ENERGY LOOP BEGINS HERE ----
304 ---         DO 87 MM=17,MAX
305 ---         EM=MM
306 ---         EM=EM-0.5
307 ---         EM2=EM*EM
308 ---         SJ=J
309 ---         EMJ=EM/SJ
310 ---         EMN2=W(1,9)/EM
311 ---         EMO2=W(2,8)/EM
312 ---         EMOX=W(3,8)/EM
313 --- C???? IONEN2 -- ENERGIES AT WHICH SECONDARIES ARE CREATED
314 --- C???? FOR N2IONS SEE GREEN & STOLARSKI
315 ---         IONEN2=MM-20
316 ---         IONEO2=MM-19
317 ---         IONEOX=MM-16
318 ---         QOFAC=QO/EM2
319 ---         N2IONS=QOFAC*AO(1,9)*EMN2**P(1,9)*(1.-EMJ**GAM(1,9))**
320 ---        1XNU(1,9)*EMJ**OMEG(1,9)
321 ---         O2IONS=QOFAC*AO(2,8)*EMO2**P(2,8)*(1.-EMJ**GAM(2,8))**
322 ---        1XNU(2,8)*EMJ**OMEG(2,8)
323 ---         OXIONS=QOFAC*AO(3,8)*EMOX**P(3,8)*(1.-EMJ**GAM(3,8))**
324 ---        1XNU(3,8)*EMJ**OMEG(3,8)
325 ---         TN2ION=TN2ION+N2IONS
326 ---         TO2ION=TO2ION+O2IONS
327 ---         TOXION=TOXION+OXIONS
328 --- C$$$$$$$$ ALT. LOOP BEGINS HERE $$$$
329 ---         DO 86 I=M1,IMAX
```

```
330 ---         SECN2P=N2IONS*ZN2(I)*(PHIUP(I)+PHIDWN(I))
331 ---         SECO2P=O2IONS*ZO2(I)*(PHIUP(I)+PHIDWN(I))
332 ---         SECOXP=OXIONS*ZOX(I)*(PHIUP(I)+PHIDWN(I))
333 ---         PRODUP(I,IONEOX)=PRODUP(I,IONEOX)+.5*SECOXP/AVMU
334 ---         PRODWN(I,IONEOX)=PRODWN(I,IONEOX)+.5*SECOXP/AVMU
335 ---         IF (IONEO2 .LE. 0) GO TO 1086
336 ---         PRODWN(I,IONEO2)=PRODWN(I,IONEO2)+.5*SECO2P/AVMU
337 ---         PRODUP(I,IONEO2)=PRODUP(I,IONEO2)+.5*SECO2P/AVMU
338 ---         IF (IONEN2 .LE. 0) GO TO 1086
339 ---         PRODUP(I,IONEN2)=PRODUP(I,IONEN2)+.5*SECN2P/AVMU
340 ---         PRODWN(I,IONEN2)=PRODWN(I,IONEN2)+.5*SECN2P/AVMU
341 ---  1086 SN2ION(I)=SN2ION(I)+SECN2P
342 ---         PSEC(I)=PSEC(I)+SECOXP
343 ---     86 SECION(I)=SECION(I)+SECN2P+SECO2P+SECOXP
344 ---     87 CONTINUE
345 --- C?$?$?$ END ENERGY AND ALTITUDE LOOPS ?$?$?$
346 ---         IF(ISW.NE.2) WRITE(6,326) TN2ION, TO2ION, TOXION
347 ---     82 J=J-1
348 ---         IF (J .LT. JMIN) GO TO 80
349 ---         GO TO 23
350 --- C////////// MAIN CALCULATIONS  END HERE ////////////
351 ---     80 CONTINUE
352 ---         ZREDEM=0.0
353 ---         DO 95 I=M1,IMAX
354 ---         EHEAT(I)=EHEAT(I)*AVMU
355 ---         SUMSUM=(SUMUP(I)+SUMDWN(I))
356 ---         SUMNET=(SUMUP(I)-SUMDWN(I))
357 ---         REDEM(I)=REDEXC(I)/(1.32+4.35E-9*ZN2(I))
358 ---         ZREDEM=ZREDEM+REDEM(I)*1.0E6
359 ---         IF(ISW.EQ.2) GO TO 95
360 ---         IF(I.EQ.M1) WRITE(6,324)
361 ---           WRITE(6,311) I,Z(I),SUMSUM,SUMNET,SUMUP(I),SUMDWN(I),EHEAT(I)
362 ---        >     ,REDEXC(I),TPROD(I),PSEC(I)
363 --- 95          CONTINUE
364 ---         IF(ISW.NE.2) WRITE(6,303)
365 ---         IF(ISW.NE.2) WRITE(6,304)  EFLUX,ETOT,EHAV
366 ---         IF(ITEST.NE.0) WRITE(6,337) ITEST
367 ---     26 FORMAT(1H ,8X,5HTN2SA,8X,5HTO2SA,7X,5HTOXSA,8X,6HN2SIGS,7X,6HO2SIG
368 ---       1S,9X,6HOXSIGS,6X,6HO1DSIG,6X,6HO1SSIG,6X,6HO3SSIG)
369 ---     27 FORMAT(1H ,8E12.3)
370 --- 303       FORMAT(/3X,*EFLUX*,5X,*ETOT*,5X,*EHAV*/)
371 ---    304 FORMAT(8E10.3)
372 ---    305 FORMAT (8F10.6)
373 ---    310 FORMAT(1H ,4X1HI,9X,1HZ,8X,6HPHIDWN,6X,5HPHIUP,7X
374 ---       1     ,*PROD*,7X,*PRODUP*,7X,*PRODWN*,7X,*EHEAT*)
375 ---    311 FORMAT(1H ,I5,F12.1,12E12.3)
376 ---    314 FORMAT (///1H1,15X,*THIS IS THE PHOTOELECTRON FLUX FOR ENERGY
377 ---       1 J=*,I3,2HEV)
378 ---    320 FORMAT(1H ,7X,1HZ,13X,6HPRODUP,11X,6HPRODWN,12X,4HPROD)
379 --- 324       FORMAT(1H  ,3X,*I*,9X,*ALT*,6X,*SUMSUM*,6X,*SUMNET*,6X,
380 ---       1       *SUMUP*,6X,*SUMDWN*,6X,*EHEAT*,6X,*REDEXC*,6X,*TPROD*)
381 ---    326 FORMAT(1H ,3E15.3)
382 ---    330 FORMAT(12I10)
383 ---    335 FORMAT(3(1PE15.4))
384 ---    336 FORMAT(8(1PE10.3))
```

```
385 --- 337   FORMAT(*ND OF PTS WITH CTS ENERGY LOSS >1 EV =*,I4)
386 ---         RETURN
387 ---          END
388 --- C.................<PSTRID>.................
389 ---         SUBROUTINE TRIS(IDIR,M2,M,J,MT,BM,BG,Z,IMAX,DH)
390 ---         DIMENSION PUD(3),A(300),B(300),C(300),D(300)
391 ---        COMMON/TRI/PROD(300),PRODWN(300,100),PRODUP(300,100),
392 ---       1     PHIDWN(300),PHIUP(300),T1(300),T2(300),DS(300)
393 ---         DIMENSION Z(300),BM(300),BG(300)
394 ---          COMMON/PANGS/ZPAS,PASK,IPAS,IPASC,FPAS
395 ---         DATA   ISW , FAC2 , JSN , JSW  , KITER , M1 ,ZPRIN ,ZZZZ
396 ---       1       / 1  ,   0  ,  3  ,  1   ,  75  ,  1, 300 ,1000 /
397 ---         IEQ=(IMAX+1)/2
398 ---        IMXM=IMAX+1-MT
399 ---        IMXM1=IMXM-1
400 ---        IMXM2=IMXM-2
401 --- CEEEE DO LOOP FOR ITERATING THE SOLUTIONS... PHIDWN IS SOLVED
402 --- CEEEE USING TRIDAG SOLVER FOR ONE HEMISPHERE, PHIUP IS SOLVED
403 --- CEEEE ANALYTICALLY TO UPPER BDY IN CONJUGATE H-S, PHIUP FOR
404 --- CEEEE C.H.S IS FOUND USING TRIDAG, THEN PHIDWN IS SOLVED ANALYTICALLY
405 --- CEEEE BACK ALONG THE FIELD LINE
406 --- C.... COEFFS FOR TRIDIAG SOLVER IMPIT ...
407 ---       DO 40 I=M2,M
408 ---         DELZ=2*DS(I)
409 ---         IF(IDIR.EQ.-1) DELZ=2*DS(I+1)
410 ---         DLB=JSW*(BM(I+1)-BM(I-1))/(BM(I)*DELZ)
411 ---         DBG=ISW*(BG(I+1)-BG(I-1))/(2*DH)
412 ---         DSLB=-JSW*4*(BM(I+1)-2*BM(I)+BM(I-1))/(DELZ**2*BM(I))
413 ---       1     -DLB*DBG+DLB*DLB
414 ---         DO 5 IK=1,3
415 ---          IJ=2-IK
416 ---         IF(IDIR.EQ.1) PUD(IK)=PRODWN(I-IJ,J)
417 --- 5       IF(IDIR.EQ.-1) PUD(IK)=PRODUP(I-IJ,J)
418 ---         PUDCON=PRODUP(I,J)
419 ---         IF(IDIR.EQ.-1) PUDCON=PRODWN(I,J)
420 ---         DPR1=IDIR*(PROD(I+1)-PROD(I-1))/(2*DELZ)
421 ---         DPR2=IDIR*(PUD(3)-PUD(1))/DELZ
422 ---        PHI=4./DELZ**2
423 ---        ALPHA=-(T1(I+1)-T1(I-1))/DELZ/T1(I)
424 ---        BETA=IDIR*(-T2(I)*ALPHA-(T2(I+1)-T2(I-1))/DELZ)-T2(I)**2+
425 ---       1     T1(I)**2
426 ---        D(I)=(PROD(I)/2)*(-T1(I)-T2(I)-ALPHA*IDIR)-DPR1
427 ---       1    +PUD(2)*(-ALPHA*IDIR-T2(I))-DPR2
428 ---       1   -PUDCON*T1(I)
429 ---         BETA=BETA+DSLB-ALPHA*DLB+DLB*DLB
430 ---         ALPHA=(ALPHA-2*DLB+DBG)/DELZ
431 ---         D(I)=D(I)+DLB*(PROD(I)/2+PUD(2))*IDIR
432 ---        C(I)=PHI+ALPHA
433 ---        B(I)=-2.*PHI+BETA
434 ---        A(I)=PHI-ALPHA
435 ---         IF(JSN.EQ.1) WRITE(6,311) I,Z(I),ALPHA,BETA,DPR1,DPR2
436 ---       1           ,A(I),B(I),C(I),D(I)
437 --- 40        CONTINUE
438 --- C.... END OF D.E. COEFFS --- SOLUTION FOR NEAR H-S ....
439 ---         IF(IDIR.EQ.-1) GO TO 10
```

```
440 ---           D(M2)=D(M2)-A(M2)*PHIDWN(M2-1)
441 ---           D(M)=D(M)-C(M)*PHIDWN(M+1)
442 ---           CALL TRIDAG(PHIDWN,M2,M,A,B,C,D)
443 --- C:::::::: PHIUP IS EVALUATED  ANALYTICALLY   ::::
444 ---         DO 60 I=M2,IMXM1
445 ---            T2M=T2(I)-.5*(BM(I+1)-BM(I-1))/(BM(I)*DS(I))*JSW
446 ---          R1=(T1(I)*PHIDWN(I)+(PROD(I)+2.*PRODUP(I,J))/2.)/T2M
447 ---          PHIUP(I)=R1+(PHIUP(I-1)-R1)*EXP(-T2M*DS(I))
448 ---           IF(I.EQ.IPAS+1) PHIUP(I)=PHIUP(I)*(1-FPAS)
449 --- 60          CONTINUE
450 ---           RETURN
451 --- C***** CONJUGATE SOLUTIONS *******
452 --- 10          D(M2)=D(M2)-A(M2)*PHIUP(M2-1)
453 ---           D(M)=D(M)-C(M)*PHIUP(M+1)
454 ---         CALL  TRIDAG(PHIUP,M2,M,A,B,C,D)
455 --- C:::::::: PHIDWN IS EVALUATED ANALYTICALLY   ::::
456 ---         DO 160 I=M1,IMXM2
457 ---           K=IMAX-I
458 ---           T2P=T2(K)+.5*(BM(K+1)-BM(K-1))/(BM(K)*DS(K+1))*JSW
459 ---          R1=(T1(K)*PHIUP(K)+(PROD(K)+2.*PRODWN(K,J))/2.)/T2P
460 ---          PHIDWN(K)=R1+(PHIDWN(K+1)-R1)*EXP(-T2P*DS(K+1))
461 ---           IF(K.EQ.IPASC-1) PHIDWN(K)=PHIDWN(K)*(1-FPAS)
462 --- 160         CONTINUE
463 ---    311 FORMAT(1H ,I5,F12.1,12E12.3)
464 ---           RETURN
465 ---            END
466 --- C-------------------------------------------------------------
467 ---           SUBROUTINE TRIDAG(DELTA,IF,L,A,B,C,D)
468 --- C.... FOR SOLVING A SYSTEM OF LINEAR SIMULTANEOUS EQUATIONS WITH A
469 --- C     TRIDIAGONAL COEFF MATRIX. THE EQNS ARE NUMBERED FROM IF TO L,
470 --- C     & THEIR  SUB-DIAG. , DAG. ,& SUPER-DIAG COEFFS. ARE STORED
471 --- C.....     IN THE ARRAYS A ,B& C
472 ---          DIMENSION A(300),B(300),C(300),D(300)
473 ---          DIMENSION ALPHA(300),DELTA(300),GAMMA(300)
474 --- C.....  COMPUTE INTERMEDIATE ARRAYS ALPHA & GAMMA
475 ---           ALPHA(IF)=B(IF)
476 ---           GAMMA(IF)=D(IF)/ALPHA(IF)
477 ---           IFP1=IF+1
478 ---           DO 1 I=IFP1,L
479 ---           ALPHA(I)=B(I)-A(I)*C(I-1)/ALPHA(I-1)
480 ---           GAMMA(I)=(D(I)-A(I)*GAMMA(I-1))/ ALPHA(I)
481 --- 1         CONTINUE
482 --- C.....  COMPUTE FINAL SOLUTION VECTOR V
483 ---           DELTA(L)=GAMMA(L)
484 ---           LAST=L-IF
485 ---           DO 2 K=1,LAST
486 ---           I=L-K
487 ---           DELTA(I)=GAMMA(I)-C(I)*DELTA(I+1)/ALPHA(I)
488 --- 2         CONTINUE
489 ---           RETURN
490 ---           END
```

TION DATASET IS ABOUT TO BE WRITTEN

```
)IT,D=PLIB,DN=RSTEM,DL
  1 --- C.........(RSTEM)...........
  2 ---        SUBROUTINE TFIJ(J,ILJ,IPR,N,TI,F,JB,JBS,V)
  3 --- C**** THIS SUBR SETS UP ERROR FNS. OF THE TIME-DEP ION AND
  4 --- C**** ELECTRON TEMPS, FOR SUBSEQUENT SOLUTION BY THE NEWTON
  5 --- C**** ITERATIVE PROCEDURE. THE SOLUTION IS OBTAINED IN ORTHOG
  6 --- C**** MAGNETIC FIELD COORDS FOR A COMPLETE DIPOLE FIELD LINE.
  7 --- C... IFLUX=ION HEAT FLUX FOR ENERGY CONSERVATION TEST
  8 --- C... KE,KI=ELECTRON,ION THERMAL CONDUCTIVITY; TE=ELECTRON TEMP
  9 --- C... UE=ELECTRON DRIFT VELOCITY.L,Q=LOSS/GAIN OF THE IONS
 10 --- C... L,Q,=LOSS/GAIN OF IONS; PPP=IMPLICIT-EXPLICIT
 11 --- C... ELECTRON AND ION DENSITIES; V=VELOCITY IN DENSITY
 12 --- C... SOLUTIONS BUT IS USED TO GET STEADY STATE ELECTRON TEMP HERE
 13 --- C... HFLX=HEAT FLUX IN EV
 14 ---           IMPLICIT REAL (A-H,K-L,N-Z)
 15 ---        REAL  Z,DT,DH,THF,EPS,IFLUX,TF
 16 ---         DIMENSION KE(3),KI(3),TE(3),UE(3)
 17 ---          DIMENSION N(4,300),TI(3,300),F(2),L(2),Q(2),PPP(3,3),V(2)
 18 ---        COMMON/VN/U(2,300),BG(300),BM(300),GR(2,300),R(2,300),SL(300)
 19 ---        COMMON/ND/DN(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
 20 ---        COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,ION,TF,ITF
 21 ---        COMMON/SAV/NSAVE(2,300),TISAV(3,300),FY(2,300),UN(300),EHT(300)
 22 ---        COMMON/FQN/JON,JLL(5),JUL(5)
 23 ---        COMMON/CPRIN/HFLX(2,300)
 24 ---          DATA   BK    ,   BOLTZ   ,JRIT,NHE,STS   ,DL3 ,DL2
 25 ---       1      / 8.63E-5,1.3807E-16,  0 , 0., 1.,-.02798 , 3./
 26 ---          DH2=2*DH
 27 ---         IPR=3
 28 --- C.... SLOSS FINDS IMPL-EXPL MIX OF ELECTRON AND ION DENSITIES
 29 --- C... STS=0 FOR STEADY STATE SOLUTION; ICNT IS A COUNTER
 30 --- C... BMK IS USED TO NORMALIZED THE HEAT FLUXES TO  1000KM ALTITUDE
 31 ---          CALL SLOSS(J,ILJ,N,TI,PPP,L,Q)
 32 ---          IF((J.EQ.2).AND.(ILJ.EQ.0)) WRITE(6,58)
 33 ---         IF(ITF.EQ.3) STS=0
 34 ---         IF(ITF.NE.3) STS=1
 35 ---          ICNT=ICNT+1
 36 ---         JMAXD2=JMAX/2
 37 ---          IF(ICNT.NE.1)  GO TO 100
 38 ---          DO 90 IK=1,JMAXD2
 39 ---          IF(Z(IK).LT.1000) IF=IK
 40 --- 90       IF(Z(IK).LT.1000) BMK=BM(IK)
 41 --- 100         CONTINUE
 42 --- C.... CONVERT EHT TO C.G.S UNITS (ERGS) KEI IS LOSS RATE FROM ELECS
 43 --- C.... TO IONS, GRADT & DIVV ARE CONVECTION TERMS FROM DENSITY PROGS
 44 ---          ERGS=EHT(J)*1.6E-12
 45 ---          NE=PPP(1,2)+PPP(2,2)+N(3,J)
 46 ---          NEV=PPP(1,2)+PPP(2,2)
 47 ---          NI=PPP(1,2)/16+PPP(2,2)+N(3,J)/30
 48 ---          KEI=1.232E-17*NE*NI*(TI(3,J)-TI(2,J))/(TI(3,J)**1.5)
 49 --- C**** HEAT LOSS DUE TO LOSS OF PARTICLES ****
 50 ---          HLQ=1.5*BOLTZ*TI(3,J)*(Q(1)+Q(2)-L(1)*PPP(1,2)-L(2)*PPP(2,2))
 51 ---          BGSQ=BG(J)**2
 52 ---          BGDBG=BG(J)*(BG(J+1)-BG(J-1))/DH2
 53 ---          BGBDB=BG(J)*(BM(J+1)-BM(J-1))/(DH2*BM(J))
 54 ---         TE2=TI(3,J)
```

```
55 ---          TNJ=TN(J)
56 ---          SQTE=SQRT(TE2)
57 --- C----IF ILJ=3, LOCAL ENERGY BALANCE FOR TE REQUIRED, JUMP TO TE SECTION
58 ---         IF(ILJ.EQ.3) GO TO 13
59 --- C..... ELECTRON AND ION THERMAL CONDUCTIVITY(KE,KI) AND DEFINE TE ...
60 --- C... REESEROBLE S.P.R (1975) P214; KE,KI,UE ARE EVALUTED AT 3 POINTS
61 --- C... TO ALLOW GRADIENTS TO BE CALC. KN GIVES THE REDUCTION OF KE
62 --- C... DUE TO NEUTRAL COLLISIONS, SEE ALSO REESEROBLE
63 --- 7          DO 12 I=1,3
64 ---           JT=J-2+I
65 ---           NET=PPP(1,I)+PPP(2,I)
66 ---           TE(I)=TI(3,JT)
67 ---          UE(I)=(FY(1,JT)+FY(2,JT))/NET
68 ---           SQTE=SQRT(TE(I))
69 ---           KN2=(2.82E-17*SQTE-3.41E-21*SQTE*TE(I))*N2N(JT)
70 ---           KO2=(2.2E-16+7.92E-18*SQTE)*O2N(JT)
71 ---           KO=3.4E-16*ON(JT)
72 ---           KHE=5.6E-16*NHE
73 ---           KH=(5.47E-15-7.45E-19*TE(I))*HN(JT)
74 ---           KN=3.22E+4*TE(I)**2*(KN2+KO2+KO+KHE+KH)/(NET+N(3,JT))
75 ---           KI(I)=1.84E-8*(PPP(1,I)+4*PPP(2,I))*TI(2,JT)**2.5/NET
76 ---          KE(I)=1.232E-6*TE(I)**2.5/(1+KN)
77 ---         IF(IRIT.EQ.1) WRITE(6,55) KN2,KO2,KO,KHE,KH,KN,KI(I),KE(I)
78 --- 12          CONTINUE
79 ---           SQTE=SQRT(TI(3,J))
80 --- C......
81 --- C....TI ERROR FNCTS I.E. ION T-DEP TEMP EQN.....
82 ---            DTE=(TI(2,J+1)-TI(2,J-1))/DH2
83 ---            DDTE=(TI(2,J+1)-2*TI(2,J)+TI(2,J-1))/(DH**2)
84 ---            IF(ILJ.EQ.0) HFLX(1,J)=-BMK*KI(2)*BG(J)*DTE/(1.6E-12*BM(J))
85 ---          TENCR=STS*1.5*BOLTZ*NE*(TI(2,J)-TISAV(2,J))/DT
86 ---           DBG=-BGDBG*KI(2)*DTE
87 ---           DSQT=-BGSQ*KI(2)*DDTE
88 ---           DKE=-(KI(3)-KI(1))*BGSQ*DTE/DH2
89 --- C         DKE=-2.5*BGSQ*KI(2)*DTE**2/TI(1,J)
90 ---           DBM=BG(J)*KI(2)*BGBDB*DTE
91 --- C,,,,   LOSS RATE COEFF TO NEUTRALS (KIN) REES & ROBLE(1975) P220 ,,,,,
92 --- C,,,,   RCE = RESONANT CHARGE EXCHANGE, POL = POLARIZATION INTERACTION
93 ---           RCE=0.21*PPP(1,2)*ON(J)+1.4*PPP(2,2)*HN(J)
94 ---           POL1=PPP(1,2)*(6.6*N2N(J)+2.8*NHE+5.8*O2N(J)+5.6*ON(J))
95 ---       1        +PPP(2,2)*(5.5*NHE+1.9*HN(J))
96 ---           POL2=(0.36*PPP(1,2)*HN(J)+0.4*PPP(2,2)*ON(J))*SQRT(TNJ)
97 ---           KIN=(RCE*SQRT(TNJ+TI(2,J))+POL1+POL2)*1.6E-26
98 ---           HLOSI=+KIN*(TI(2,J)-TNJ)
99 ---           GRADT=1.5*NEV*BOLTZ*BG(J)*DTE*UE(2)
00 ---           DIVV=NEV*BOLTZ*TI(2,J)*((UE(3)-UE(1))*BG(J)/DH2-BGBDB)
01 ---           F(1)=TENCR+DBG+DSQT+DKE+DBM-KEI+HLOSI+GRADT+DIVV
02 --- C.....
03 --- C.... END TI ERR FNS WRITE VARIABLES IF DESIRED.....
04 ---           IF(ILJ.NE.0)  GO TO 5
05 ---          IF(ITER.NE.3) GO TO 5
06 ---         IF(J.LT.JLL(1).OR.J.GT.JUL(1))GO TO 5
07 ---           WRITE(6,56) J,Z(J)
08 ---           IF(J.EQ.2) WRITE(6,50)
09 ---           IF((J/9)*9.EQ.J) WRITE(6,50)
```

```
110 ---           WRITE(6,55) F(1),TENCR,DBG,DSQT,DKE,DBM,KEI,HLOSI,GRADT
111 --- 5         CONTINUE
112 --- C,,,,,,,
113 --- C,,,,,,,,,  ELECTRON TEMP ERR FNS ,,,,,,,
114 ---            DTE=(TI(3,J+1)-TI(3,J-1))/DH2
115 --- C,,,, FLUX=HEAT FLUX IN EV NORMALIZED TO LOWEST LEVEL ,,,,
116 ---            IF(ILJ.EQ.0) HFLX(2,J)=-BMK*KE(2)*BG(J)*DTE/(1.6E-12*BM(J))
117 ---            DDTE=(TI(3,J+1)-2*TI(3,J)+TI(3,J-1))/(DH**2)
118 --- C-----  ELEC-NEUT ELAS COLL LOSS RATES SEN(1978) REV GEOPHYS P366
119 ---  13        TDIF=TE2-TNJ
120 ---           LEN2=1.77E-19*NE*N2N(J)*(1-1.2E-4*TE2)*TE2*TDIF
121 ---           LEO2=1.21E-18*NE*O2N(J)*(1+3.6E-2*SQTE)*SQTE*TDIF
122 ---           LEO=7.9E-19*NE*ON(J)*(1+5.7E-4*TE2)*SQTE*TDIF
123 ---           LEH=9.63E-16*NE*HN(J)*(1-1.35E-4*TE2)*SQTE*TDIF
124 --- C###  ROTATIONAL LOSS RATES BEK 268 ####
125 ---           LRN2=2.0E-14*NE*N2N(J)*TDIF/SQTE
126 ---           LRO2=7.0E-14*NE*O2N(J)*TDIF/SQTE
127 --- C+++  N2 VIB LOSS RATES BEK P268 AND SEN P364  +++++
128 ---           EF=1.06E+4+7.51E+3*TANH(1.1E-3*(TE2-1800))
129 ---           GE=3300+1.233*(TE2-1000)-2.056E-4*(TE2-1000)*(TE2-4000)
130 ---           LVN2=-2.99E-12*NE*N2N(J)*EXP(EF*(TE2-2000)/
131 ---          1   (2000*TE2))*(EXP(-GE*TDIF/(TE2*TNJ))-1)
132 --- C$$$$ O2 VIB LOSS RATE SEN P364  $$$$
133 ---           HS=3300-839*SIN(1.91E-4*(TE2-2700))
134 ---           LVO2=-5.196E-13*NE*O2N(J)*EXP(HS*(TE2-700)/(700*TE2))
135 ---          1         *(EXP(-2770*TDIF/(TE2*TNJ))-1)
136 --- C
137 --- C;;;;;  FINE STRUCTURE EXCITATIONS SEN P365 ;;;;;;
138 --- C;;;;; NOTE THAT D1,D2,E1 ETC. MAY NEED TO BE CHANGED
139 --- C;;;; NOTE THAT A TERM IS ADDED TO NE AT LOW ALTS FOR O2+ , NO+
140 ---           D1=EXP(-228/TNJ)
141 ---           D2=EXP(-326/TNJ)
142 ---           E1=EXP(-228/TE2)
143 ---           E2=EXP(-326/TE2)
144 ---           E3=EXP(-98/TE2)*D1
145 ---           LF1=8.49E-6*TE2**0.519*(0.02*(D1-E1)-5.91E-9*
146 ---          1      TDIF*(2.019*D1+(228/TE2+2.019)*E1))
147 ---           LF2=7.7E-6*TE2**0.3998*(0.028*(D2-E2)-5.91E-9*
148 ---          1      TDIF*(1.8998*D2+(326/TE2+1.8998)*E2))
149 ---           LF3=2.22E-7*TE2**0.768*(0.008*(D2-E3)-5.91E-9*
150 ---          1      TDIF*(2.268*D2+(98/TE2+2.268)*E3))
151 ---           ZFO=5+3*D1+D2
152 ---           LFO=-8.629E-6*(NE+0.E8*EXP(-.05*Z(J)))*ON(J)*(LF1+LF2+LF3)/ZFO
153 --- C++++  FINE STRUCTURE OF O (D STATE) SEN P365 ++++++
154 ---           DE=2.4E+4+0.3*(TE2-1500)-1.947E-5*(TE2-1500)*(TE2-4000)
155 ---           LF1D=-1.57E-12*NE*ON(J)*EXP(DE*(TE2-3000)/(3000*TE2))
156 ---          1        *(EXP(-22713*TDIF/(TE2*TNJ))-1)
157 ---           IF(ILJ.EQ.3) GO TO 14
158 --- C,,,,  END OF ELECTRON-NEUTRAL LOSS RATES ,,,,
159 ---           TENCR=STS*1.5*BOLTZ*NE*(TI(3,J)-TISAV(3,J))/DT
160 ---            DBG=-BGDBG*KE(2)*DTE
161 ---            DSQT=-BGSQ*KE(2)*DDTE
162 ---            DKE=-(KE(3)-KE(1))*BGSQ*DTE/DH2
163 ---            DBM=BG(J)*KE(2)*BGBDB*DTE
164 ---  14        KEN=LEN2+LEO2+LEO+LEH+LRN2+LRO2
```

```
165 ---           HLOSS=(KEN*LVN2+LVO2+LFO+LF1D)*1.6E-12
166 --- C......... V(1) IS FORMED FOR BDY CDTN AND PREDICTOR ..........
167 ---           V(1)=ERGS-HLOSS-KEI
168 ---           IF(ILJ.EQ.3) RETURN
169 ---           GRADT=1.5*NEV*BOLTZ*BG(J)*DTE*UE(2)
170 ---           DIVV=NEV*BOLTZ*TI(3,J)*((UE(3)-UE(1))*BG(J)/DH2-BGBDB)
171 ---           F(2)=TENCR+DBG+DSQT+DKE+DBM+KEI+HLOSS+GRADT+DIVV-ERGS
172 --- C.....
173 --- C.... END TE ERR FNS WRITE VARIABLES ....
174 ---           IF(ILJ.NE.0)  GO TO 15
175 ---          IF(ITER.NE.3) GO TO 15
176 ---         IF(J.LT.JLL(1).OR.J.GT.JUL(1))GO TO 15
177 ---           WRITE(6,56) J,Z(J)
178 ---           WRITE(6,55) F(2),TENCR,DBG,DSQT,DKE,DBM,KEI,HLOSS,DIVV,ERGS
179 ---           IF(Z(J).GT.1000) GO TO 15
180 ---           IF((J/5)*5.EQ.J)WRITE(6,57)
181 ---           WRITE(6,55) LEN2,LEO2,LEO,LEH,LRN2
182 ---        1    ,LRO2,LVN2,LVO2,LFO,LF1D
183 --- 15        CONTINUE
184 --- C... THIS SECTION TESTS CONSERVATION OF ENERGY ABOVE 1000KM
185 --- C... THE HEAT FLUX AT 1000KM IS DETERMINED FROM INTEGRATING THE
186 --- C... PRODUCTION AND LOSS AND ALSO DIRECTLY FROM TEMP GRADIENT
187 ---           IF(ILJ.NE.0) RETURN
188 ---           IF(STS.NE.0) RETURN
189 ---           IF(J.EQ.10) PHITI=0
190 ---           IF(J.EQ.10) PHIT=0
191 ---           IF(J.LT.IF) RETURN
192 ---           PHITI=PHITI+DH*(KEI-HLOSI)/(BM(J)*BG(J))
193 ---           PHII=PHITI*BM(J)/1.6E-12
194 ---           IFLUX=HFLX(1,IF)+ABS(HFLX(1,JMAX+1-IF))
195 ---           PHIT=PHIT+DH*(ERGS-KEI-HLOSS)/(BM(J)*BG(J))
196 ---           PHI=PHIT*BM(J)/1.6E-12
197 ---           EFLUX=HFLX(2,IF)+ABS(HFLX(2,JMAX+1-IF))
198 ---           IF(J.EQ.JMAX+1-IF) WRITE(6,55) PHI,EFLUX,PHII,IFLUX
199 ---         RETURN
200 --- 50        FORMAT(9X,*F*,8X,*TENCR*,8X,*DBG*,8X,*DSQT*
201 ---        1     ,9X,*DKE*,8X,*DBM*,8X,*KEI*,8X,*HLOSS*)
202 --- 55          FORMAT(1P19E12.2)
203 --- 56        FORMAT(I5,F9.0)
204 --- 57        FORMAT(8X,*LEN2*,8X,*LEO2*,8X,*LEO*,8X,*LEH*,8X,*LRN2*,8X
205 ---        1    ,*LRO2*,8X,*LVN2*,8X,*LVO2*,8X,*LFO*,8X,*LF1D*)
206 --- 58       FORMAT(*ENTER TFIJ*)
207 ---          END
208 --- C*SLOSS**SLOSS**SLOSS**SLOSS**SLOSS**SLOSS**SLOSS**SLOSS*
209 ---         SUBROUTINE  SLOSS (J,JLI,N,TI,PPP,L,Q)
210 --- C.... SUBROUTINE SLOSS CALCULATES ION PROD. & LOSS....
211 --- C... THE PROGRAM ALSO DETERMINES THE IMPLICIT-EXPLICIT ION DENSITY
212 ---             IMPLICIT REAL (A-H,L,N-Z)
213 ---         REAL  Z,DT,DH,THF,EPS
214 ---           DIMENSION N(4,300),TI(3,300),F(2),L(2),Q(2),PPP(3,3),V(2)
215 ---         COMMON/ND/ON(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
216 ---         COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,ION,TF,ITF
217 ---         COMMON/SAV/NSAVE(2,300),TISAV(3,300),FY(2,300),UN(300),EHT(300)
218 --- C==================FOR CRANC-NICOLSON==================
219 --- C;;;;;;;  CALCULATE DENSITIES AND VELOCITIES
```

```
220 ---           IONP1=ION+1
221 ---            DO 409 I=1,3
222 --- 409        PPP(IONP1,I)=0
223 ---           JON=1
224 ---            DO 410 I=1,ION
225 ---         PPP(I,1)=(1-THF)*NSAVE(I,J-1)+THF*N(I,J-1)
226 ---         PPP(I,2)=(1-THF)*NSAVE(I,J)+THF*N(I,J)
227 ---         PPP(I,3)=(1-THF)*NSAVE(I,J+1)+THF*N(I,J+1)
228 ---         PPP(IONP1,1)=PPP(IONP1,1)+PPP(I,1)
229 ---         PPP(IONP1,2)=PPP(IONP1,2)+PPP(I,2)
230 ---         PPP(IONP1,3)=PPP(IONP1,3)+PPP(I,3)
231 ---    410 CONTINUE
232 --- C
233 --- C+++++     CALCULATE *Q* AND *L* THE SOURCE AND LOSS TERMS
234 ---          CALL RATES(J,TI,TN,R1,R2,R3,R4,R5,R6,R7,R8)
235 ---           AL1=1.23E-4*PPP(IONP1,2)/TI(3,J)
236 ---           AL2=6.6E-5*PPP(IONP1,2)/TI(3,J)
237 --- C
238 ---         L(1)=R2*HN(J)*(R3+R4)/(1+R3/AL2+R4/AL1)
239 ---         L(2)=R1*ON(J)
240 ---         Q(1)=(R1*ON(J)*PPP(2,2)+PHION(J))
241 ---         Q(2)=(R2*HN(J)*PPP(1,2))
242 ---              RETURN
243 ---                END

ATION DATASET IS ABOUT TO BE WRITTEN

GIVEN,  RF DONE
ATION DATASET HAS NOW BEEN WRITTEN

-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    26/100 SECONDS IS ELAPSED TI
```

```
DIT,D=PLIB,DN=RSTER,OL
  1 --- C.............<RSTER>.....................
  2 ---       SUBROUTINE TERD(J,QB,QJ,QA,QL,QM,QU,LOC)
  3 --- C---- REAL LINEAR INTERPOLATION -----
  4 ---         IMPLICIT REAL(A-H,O-Z)
  5 ---       COMMON/DELZ/HL,HU,IT
  6 --- C..... QL AND QU ARE INTERPOLATION POINTS
  7 --- C..... OF Q IN THE (ZB,ZJ) AND (ZJ,ZA)INTERVALS RESPECTIVELY
  8 --- C..... QM IS THE AVERAGE VALUE OF Q IN THE (ZJ-YL,ZJ+YL) INTERVAL
  9 ---       QL=.5*(QB+QJ)
 10 ---       QU=.5*(QJ+QA)
 11 ---       QM=.25*((QL+QJ)*HL+(QU+QJ)*HU)
 12 ---       RETURN
 13 ---         END
 14 --- C,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+
 15 ---         SUBROUTINE TERI(J,QB,QJ,QA,QL,QM,QU,LOC)
 16 ---       COMMON/DELZ/HL,HU,IT
 17 --- C.....    EXPONENTIAL INTERPOLATION
 18 ---       IF((QB*QJ.GT.0).AND.(QJ*QA.GT.0))GO TO 100
 19 ---       WRITE(6,600)
 20 ---   600 FORMAT(*TERD*)
 21 --- C
 22 ---   100 QL=SQRT(QB*QJ)
 23 ---       QU=SQRT(QJ*QA)
 24 ---       QM=.5*HU*(QJ-QU)/ALOG(QJ/QU)
 25 ---      >    +.5*HL*(QL-QJ)/ALOG(QL/QJ)
 26 ---       RETURN
 27 ---       END
 28 --- C,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+
 29 ---         SUBROUTINE DAVE(ION,J,ANL,ANM,ANU,PL,PM,PU,N,NSAVE)
 30 --- C....     CALCULATES ANTE AND POST VALUES
 31 --- C....     OF HALF INTERVAL, AND AVERAGE
 32 --- C....     OF (ION DENSITY)/(MAGNETIC FIELD)
 33 ---         IMPLICIT REAL(A-H,N-Z)
 34 ---       COMMON/VN/U(2,300),BG(300),BM(300),GR(2,300),GP(2,300),SL(300)
 35 ---       DIMENSION ANL(2),ANM(2),ANU(2),PL(2),PM(2),PU(2),N(4,300)
 36 ---      >    ,NSAVE(2,300)
 37 --- C
 38 ---       DO 100 I=1,ION
 39 --- C....     ANTE VALUES
 40 ---       BB=NSAVE(I,J-1)/BM(J-1)
 41 ---       C=NSAVE(I,J)/BM(J)
 42 ---       A=NSAVE(I,J+1)/BM(J+1)
 43 ---       IF((A.LT.0.0).OR.(C.LT.0.0))WRITE(6,115)
 44 --- 115     FORMAT(*NEGATIVE ANTE*)
 45 ---       CALL TERI(J,BB,C,A,QL,QM,QU,3)
 46 ---       ANL(I)=QL
 47 ---       ANM(I)=QM
 48 ---       ANU(I)=QU
 49 --- C....     POST VALUES
 50 ---       BB=N(I,J-1)/BM(J-1)
 51 ---       C=N(I,J)/BM(J)
 52 ---       A=N(I,J+1)/BM(J+1)
 53 ---       IF((A.LT.0.0).OR.(C.LT.0.0))WRITE(6,116)
 54 --- 116     FORMAT(*NEGATIVE POST*)
```

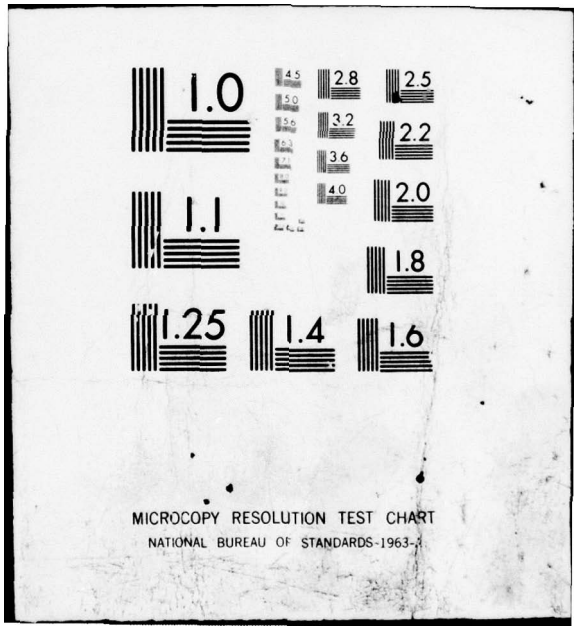
```
      CALL TERI(J,BB,C,A,QL,QM,QU,4)
      PL(I)=QL
      PM(I)=QM
      PU(I)=QU
  100 CONTINUE
      RETURN
      END
C,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+
      SUBROUTINE CHEMO(ION,JI,SOURCE,N,NSAVE,TI,THF)
C.... THIS PROGRAM DETERMINES THE INTERPOLATED PRODUCTION AND LOSS
C.... PROCESSES. IT CALLS RATES TO GET THE RATE CONSTANTS AND TERD
C.... TO DO THE INTERPOLATION
       IMPLICIT REAL(A-H,L,N-Z)
        REAL THF
        DIMENSION N(4,300),TI(3,300),NSAVE(2,300),Q(2,3),L(2,3)
     >   ,SOURCE(2),SINK(2),DML(2),DMU(2),CN(2,3)
      COMMON/VN/U(2,300),BG(300),BM(300),GR(2,300),GP(2,300),SL(300)
      COMMON/ND/ON(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
      COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,FHT,ITER,EPS,NION,TF,ITF
      COMMON/FGN/JON,JLL(5),JUL(5)
      THP=1.0-THF
      DO 101 I=1,ION
      CN(I,1)=THP*NSAVE(I,JI-1)+THF*N(I,JI-1)
      CN(I,2)=THP*NSAVE(I,JI)+THF*N(I,JI)
  101 CN(I,3)=THP*NSAVE(I,JI+1)+THF*N(I,JI+1)
C
      DO 300 K=1,3
      J=K+JI-2
       CALL RATES(J,TI,TN,R1,R2,R3,R4,R5,R6,R7,R8)
C@@@@ SOURCES AND SINKS FOR EACH ION ARE SUMMED TOGETHER TO FORM Q(I,J)
      L(1,K)=(R2*HN(J)+R3*N2N(J)+R4*O2N(J))*CN(1,K)
      L(2,K)=R1*ON(J)*CN(2,K)
      Q(1,K)=(L(2,K)+PHION(J)-L(1,K))/BM(J)
      Q(2,K)=(R2*HN(J)*CN(1,K)-L(2,K))/BM(J)
C-------- PRINTING OF INDIVIDUAL PRODUCTION AND LOSS TERMS -----
       IF(JI.LT.JLL(2).OR.JI.GT.JUL(2)) GO TO 300
       IF(K.NE.2) GO TO 300
       IF(ITER.NE.4) GO TO 300
       IF(JI.EQ.2) WRITE(6,11)
       IF((JI/30)*30.EQ.JI) WRITE(6,11)
       R2HN=R2*HN(J)
       R3N2=R3*N2N(J)
       R4O2=R4*O2N(J)
       R1ON=R1*ON(J)
       WRITE(6,13) JI,Z(JI),R2HN,R3N2,R4O2,R1ON,CN(1,K),CN(2,K),PHION(J)
   11  FORMAT(/2X,*J*,5X,*ALT*,7X,*R2HN*,8X,*R2N2*,8X,*R4O2*,8X,*R1ON*
     >      ,8X,*N(O+)*,8X,*N(H+)*,8X,*PHION*)
   13  FORMAT(I5,F9.0,1P12E12.4)
  300 CONTINUE
C.... TERD IS CALLED TO INTERPOLATE
      DO 400 I=1,ION
      QB=Q(I,1)
      QJ=Q(I,2)
      QA=Q(I,3)
C       WRITE(6,350) I,JI,QB,QJ,QA,QL,QM,QU
```

AD-A073 892 MICHIGAN UNIV ANN ARBOR F/G 4/1
A SIMULATION OF THE MIDLATITUDE PLASMASPHERE AND IONOSPHERE.(U)
MAY 79 E R YOUNG, D G TORR, P RICHARDS F19628-77-C-0007
AFGL-TR-79-0125 NL

UNCLASSIFIED

2 OF 2
AD
A073892

END
DATE
FILMED
10-79
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
110 ---         CALL TERD(JI,QB,QJ,QA,QL,QM,QU,5)
111 --- C        WRITE(6,350) I,JI,QB,QJ,QA,QL,QM,QU
112 --- 350     FORMAT(2I4,12E12.2)
113 ---         SOURCE(I)=QM
114 ---  400    CONTINUE
115 ---         RETURN
116 ---         END
117 --- C*£$#%@,?!*£$#%@,?!*£$#%@,?!*£$#%@,?!*£$#%@,?!*£$#%@,?!*£$#%@,
118 ---         SUBROUTINE       HDDEQ(J,JM1,JMAX,N,TI,R,SL)
119 --- C.... CALCULATES DYNAMIC EQUILIBRIUM OF H+ AND O+ AT PT JM1 FROM
120 --- C.... DENSITIES AT LOWER BOUNDARY JM1, USING GEOPOTENTIAL GP(J)-GP(JM1)
121 --- C.... DRAG(I) IS A TERM TAKING COLLISIONS INTO ACCOUNT. IT MAY BE
122 --- C.... INCLUDED LATER IN PLACE OF , OR IN ADDITION TO QION(I)
123 ---           IMPLICIT REAL(A-H,N-Z)
124 ---          REAL BK,MG,M
125 ---         DIMENSION M(2),O(2),E(2),RT(2),DP(2),N(4,300),TI(3,300),PP(3)
126 ---        >    ,V(2),FLUX(2),R(2,300),SL(300)
127 ---         COMMON/DEQ/QION(2),DRAG(2),GP(2,300)
128 ---          DATA     AMU     ,        BK     ,        MG    , GP(2,1) , TF , AUG
129 ---        >    / 1.6726E-24 , 1.3807E-16 , 3.9765E+20  , 0.0,  1 ,  1 /
130 ---         DATA M/16.0,1.0/,MT/1/, T2SAV/0.0/
131 --- C... EVALUATE GEOPOTENTIAL FOR DYNAMIC EQUILIBRIUM
132 --- C... ONLY IF ION TEMP CHANGES
133 ---           JEQ=(JMAX+1)/2
134 ---           IF(T2SAV.EQ.TI(2,JEQ)) GO TO 10
135 ---           DO 5 I=2,JMAX
136 ---         TH=.5*(TI(2,I-1)+TI(2,I))
137 ---         GP(1,I)=MG*(.5*AMU/BK)*(R(1,I-1)-R(2,I-1))
138 ---        >*(1./R(1,I-1)**2/TH+1./R(2,I-1)**2/TI(2,I-1))+GP(2,I-1)
139 ---         GP(2,I)=GP(1,I)+MG*(.5*AMU/BK)*(R(2,I)-R(1,I-1))
140 ---        >*(1./R(1,I-1)**2/TH+1./R(2,I)**2/TI(2,I))
141 --- 5         CONTINUE
142 --- 10         CONTINUE
143 ---           T2SAV=TI(2,JEQ)
144 ---           IF(J.EQ.1) RETURN
145 --- C... QION IS ZEROED AT PRESENT, MAY BE USED LATER
146 ---           QION(1)=0
147 ---           QION(2)=0
148 --- C....
149 ---         NEJM1=N(1,JM1)+N(2,JM1)
150 ---         O3=NEJM1*TI(3,JM1)/TI(3,J)
151 ---         DO 1 I=1,2
152 ---         RT(I)=(TI(3,J)+TI(3,JM1))/(TI(2,J)+TI(2,JM1))
153 ---         E(I)=EXP(-M(I)*(GP(2,J)-GP(2,JM1)))*EXP(QION(I)*(SL(J)-SL(J-1)))
154 ---         O(I)=N(I,JM1)*TI(2,JM1)*E(I)/TI(2,J)
155 ---         DP(I)=RT(I)/(RT(I)+1.0)-.5
156 ---       1 CONTINUE
157 ---         N(1,J)=O(1)*SQRT(O3/(O(1)+O(2)))*((O3/(O(1)+O(2)))**DP(1))
158 ---         N(2,J)=O(2)*SQRT(O3/(O(1)+O(2)))*((O3/(O(1)+O(2)))**DP(2))
159 ---          RETURN
160 ---         END
161 --- C,!"#$%£*+?,!"#$%£*?,!"#$%£*?,!"#$%£*?,!"#$%£*?,!"#$%£*?
162 ---         SUBROUTINE HOEQ(J,JMAX,DT,THF,DCR,N,TI,NSAVE)
163 --- CC....  FINDS THE NEW CHEMICAL EQUILIBRIUM DENSITIES
164 --- CC....  OF O+ AND H+ AT POINT J FOR FRACTION IMPLICITNESS
```

```
165 --- CC.... IT ALSO CALCULATES TE FROM LOCAL HEATING=COOLING
166 ---         IMPLICIT REAL(A-H,L,N-Z)
167 ---          REAL DT,THF
168 ---        COMMON/ND/ON(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
169 ---          DIMENSION L(2,2),N(4,300),TI(3,300),NSAVE(2,300),V(2),F(2)
170 ---         CALL RATES(J,TI,TN,R1,R2,R3,R4,R5,R6,R7,R8)
171 --- C.... EVALUATE LOSS RATES
172 ---        L(1,1)=(R2*HN(J)+R3*N2N(J)+R4*O2N(J))
173 ---        L(1,2)=+R1*ON(J)
174 ---        L(2,1)=+R2*HN(J)
175 ---        L(2,2)=R1*ON(J)
176 --- C**** CALCULATE O+ & H+ DENSITIES FROM PHOTOCHEMICAL EQUILIBRIUM
177 ---         N(1,J)=PHION(J)/L(1,1)+NSAVE(1,J)*EXP(-L(1,1)*DT)
178 ---         N(2,J)=N(1,J)*L(2,1)/L(2,2)
179 --- C.... LOWER BOUNDARY FOR TE USING NEWTON TO SOLVE LOSS=GAIN
180 --- C,,,, LOCAL HEATING=LOCAL COOLING IS TRANSFERRED FROM TFIJ VIA V(1)
181 --- C,,,, H=INCREMENT FOR EVALUATING DERIVATIVE OF V(1): DEX=DERIV OF V(1)
182 --- C,,,, FEX=V(1). ION TEMP IS SET EQUAL TO NEUTRAL TEMP. NOTE THAT THE
183 ---           H=0.0001*TI(3,J)
184 --- 20        CALL TFIJ(J,3,IPT,N,TI,F,1,1,V)
185 ---           FEX=V(1)
186 ---           TI(3,J)=TI(3,J)+H
187 ---           CALL TFIJ(J,3,IPT,N,TI,F,1,1,V)
188 ---           DEX=(V(1)-FEX)/H
189 ---           TI(3,J)=TI(3,J)-H-FEX/DEX
190 ---           IF(ABS(FEX/(DEX*TI(3,J))).GT.1.0E-3) GO TO 20
191 ---         IF((J.NE.1).OR.(J.NE.JMAX)) RETURN
192 ---         TI(1,J)=TN(J)
193 ---           TI(2,J)=TN(J)
194 ---        RETURN
195 ---        END
196 --- C*PRINT**PRINT**PRINT**PRINT**PRINT**PRINT**PRINT**PRINT*
197 ---        SUBROUTINE PRINT(IPMX,IMOD,Z,JMAX,N,TI,ITAU,ITER,IDSC,U,FY,A)
198 --- C.... THIS ROUTINE PRINTS N AND U ARRAYS ....
199 ---           IMPLICIT REAL (A-H,L,N-Z)
200 ---        REAL  Z,DT,DH,THF,EPS
201 ---         DIMENSION A(4,300),N(4,300),TI(3,300),Z(300),FY(2,300),U(2,300)
202 ---        COMMON/CPRIN/HFLX(2,300)
203 ---           IW300=0
204 ---        WRITE(IDSC,200) ITER,ITAU
205 ---   200  FORMAT(20X,*ITER=*,I3,5X,*TIME=*,I6,3X,*SECS*//)
206 ---        WRITE(IDSC,103)
207 ---   103  FORMAT(1X,*ALTIT*,4X,*J*,7X,*N(O+)*,8X,*N(H+)*,8X,*TI*,8X,*TE*
208 ---       1   ,4X,*O+ VEL*,4X,*H+ VEL*,4X,*FY1*,4X,*FY2*,4X,*HFLX1*,4X
209 ---       1   ,*HFLX2*/)
210 --- C+++++++++++++++
211 ---        DO 100 J=1,IPMX
212 ---           IF(J.EQ.1)  GO TO 102
213 ---         IF(Z(J).LT.0.0) GO TO 102
214 ---        IF(MOD(J,IMOD) .EQ. 0) GO TO 102
215 ---           GO TO 100
216 --- 102       CONTINUE
217 ---           IW300=IW300+1
218 ---           IF((IW300/30)*30.EQ.IW300) WRITE(IDSC,103)
219 ---        WRITE(IDSC,101) Z(J),J,N(1,J),N(2,J),TI(2,J),TI(3,J)
```

```
220 ---       1  ,U(1,J),U(2,J),FY(1,J),FY(2,J),HFLX(1,J),HFLX(2,J),A(1,J),A(2,J)
221 ---  101  FORMAT(F9.0,I4,1P2E13.3,0P2F10.1,1P12E9.1)
222 ---  100  CONTINUE
223 ---       RETURN
224 ---       END
225 --- C!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>
226 ---         SUBROUTINE CMINOR(J,N,TI,ENOP,ENO2P)
227 --- C.... THIS PROGRAM CALCULATES THE MINOR ION CONCENTRATIONS FROM
228 --- C.... CHEMICAL EQUILIBRIUM ; ENOP=N(NO+), ENO2P=N(O2+)
229 ---         IMPLICIT REAL(A-H,N-Z)
230 ---         COMMON/ND/ON(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
231 ---         DIMENSION N(4,300),TI(3,300)
232 ---         B=N(1,J)+N(2,J)
233 ---         CALL RATES(J,TI,TN,R1,R2,R3,R4,R5,R6,R7,R8)
234 ---         C=(R3*N2N(J)/R5+R4*O2N(J)/R6)*N(1,J)
235 ---         NE=(B+SQRT(B**2+4.0*C))/2.0
236 ---         ENOP=R3*N2N(J)*N(1,J)/(NE*R5)
237 ---         ENO2P=R4*O2N(J)*N(1,J)/(NE*R6)
238 ---         RETURN
239 ---         END
240 --- C!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>!%*?>
241 ---         SUBROUTINE RATES(J,TI,TN,R1,R2,R3,R4,R5,R6,R7,R8)
242 --- C**** THIS PROGRAM EVALUATES THE RATE CONSTANTS FOR VARIOUS REACTIONS
243 --- C**** R1,R2 APPLY TO O+ + H REACTION IN BOTH DIRECTIONS ; R3=REACTION
244 --- C**** OF O+ WITH N2 ; R4= O+ WITH O2 ; R5=REACTION OF NO+ WITH ELECS
245 --- C**** R6=REACTION OF O2+ WITH ELECS : RATES FROM TORR AND TORR 1978
246 --- C**** R7,R8 ARE FOR FUTURE USE
247 ---         DIMENSION TI(3,300),TN(300)
248 --- C... O+ + H LOSS RATES
249 ---         R1=3.8E-11*SQRT(TI(2,J)+TN(J)/16.0)
250 ---         R2=5.24E-11*SQRT(TN(J)+TI(1,J)/16.0)
251 --- C... O+(4S) LOSS TO N2 AND O2 ; TORR @ TORR 1978
252 ---         T13=(4.*TN(J)+7.*TI(1,J))/11.0
253 ---         T14=(1.*TN(J)+2.*TI(1,J))/3.
254 ---         IF(T13.GT.1700)GO TO 10
255 ---         R3=1.533E-12-5.92E-13*(T13/300.)
256 ---        >  +8.60E-14*(T13/300.)**2
257 ---         GO TO 20
258 ---   10    R3=2.73E-12 -1.15E-12*(T13/300.)
259 ---        >    +1.483E-13*(T13/300.)**2
260 ---   20    R4=2.82E-11 -7.74E-12*(T14/300.) +1.073E-12*(T14/300.)**2
261 ---        >    -5.17E-14*(T14/300.)**3 + 9.65E-16*(T14/300.)**4
262 ---         R5=4.2E-7*(300./TI(3,J))**.85
263 ---         R6=1.6E-7*(300./TI(3,J))**.55
264 ---         RETURN
265 ---         END

ATION DATASET IS ABOUT TO BE WRITTEN

GIVEN,  RF DONE
ATION DATASET HAS NOW BEEN WRITTEN

-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    27/100 SECONDS IS ELAPSED T.
```

```
DIT,D=PLIB,DN=RSDEN,DL
  1 --- C..............<RSDEN>.....................
  2 ---        SUBROUTINE DFIJ(J,JSJ,IPR,N,TI,F,JB,JBS,V)
  3 --- C-------------------------------------------------------------K
  4 --- C    THIS PROGRAM WAS WRITTEN BY EUGENE YOUNG(1978). IT IS RESPONSIBLE FOR
  5 --- C    SETTING UP THE 'ERROR FUNCTIONS' FOR THE DENSITY EQUATIONS. IT CALLS
  6 --- C    SUBR VEL TO GET VELOCITIES, HODEQ TO FILL IN THE DYNAMIC EQUIL REGION,
  7 --- C    CHEMO FOR CHEMICAL SOURCES AND SINKS AND DAVE FOR INTERPOLATED DN/DT
  8 --- C    IT INTEGRATES THE CONTINUITY EQUATIONS TO OBTAIN FLUXES.
  9 --- C-------------------------------------------------------------K
 10 ---          IMPLICIT REAL(A-H,L,N-Z)
 11 ---          REAL Z,DT,DH,THF,EPS,MASS,MG,MAGM,TF
 12 ---          INTEGER ION,NEQ
 13 ---         DIMENSION MASS(2),QSIGN(2),VL(2),VU(2),FLU(2),FLL(2),QID(2)
 14 ---       >    ,ANL(2),ANM(2),ANU(2),PL(2),PM(2),PU(2),CL(3),CM(3),CU(3)
 15 ---       >    ,Q(2),L(2),TINCR(2),V(2),FYS(2)
 16 ---           DIMENSION N(4,300),TI(3,300),F(2)
 17 ---         COMMON/VN/U(2,300),BG(300),BM(300),GR(2,300),R(2,300),SL(300)
 18 ---         COMMON/ND/ON(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
 19 ---         COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,ION,TF,ITF
 20 ---         COMMON/SAV/NSAVE(2,300),TISAV(3,300),FY(2,300),UN(300),EHT(300)
 21 ---         COMMON/FON/JON,JLL(5),JUL(5)
 22 ---         COMMON /DELZ/HLL,HUU,IT
 23 ---         COMMON/DEQ/QION(2),DRAG(2),GP(2,300)
 24 ---         DATA QSIGN/-1.,1./,AMU/1.6726E-24/,BK/1.3807E-16/
 25 --- C.... CALC DENSITIES AND VELOCITIES FOR TIME T+THF*DT
 26 --- C.... AT POINTS  J,J-1/2 AND J+1/2
 27 ---          THP=1.0-THF
 28 ---          JBP1=JB+1
 29 ---          IPR=2
 30 ---          JON=JSJ
 31 ---          IF(ITER.LT.4) JON=1
 32 ---          IF((J.EQ.2).AND.(JSJ.EQ.0)) WRITE(6,116)
 33 --- 116      FORMAT(*ENTER DFIJ*)
 34 ---         IF(J.EQ.JB.OR.J.EQ.JBS+1)GO TO 110
 35 --- C+++++ STARTING FLUX FOR CONTINUITY EQN FLUX CALCULATION
 36 ---          IF(J.LE.JB) FYS(1)=(FLL(1)+FLU(1))/2
 37 ---          IF(J.LE.JB) FYS(2)=(FLL(2)+FLU(2))/2
 38 ---         GO TO 111
 39 --- C.... HODEQ IS CALLED TO FILL IN THE DYN EQUIL REGION
 40 --- 110    DO 100 JA=JBP1,JBS
 41 ---         CALL HODEQ(JA,JA-1,JMAX,N,TI,R,SL)
 42 ---    100 CONTINUE
 43 --- C.... VEL IS CALLED TO OBTAIN THE VELOCITIES(FLUXES) AT THE LOWER 1/2 PT
 44 ---    111 CALL VEL(J-1,CL,VL,FLL,N,TI,JB)
 45 ---         DO 350 I=1,ION
 46 ---         TINCR(I)=0.0
 47 ---         Q(I)=0.0
 48 ---    350 F(I)=0.0
 49 ---         JU=J
 50 ---         IF(J.EQ.JB)JU=JBS
 51 --- C---- CALL CHEMO TO EVALUATE THE SOURCE+SINK TERM AND CALL DAVE TO EVALUATE
 52 --- C---- THE TME DERIVATIVE. BOTH CALLS ARE MADE ONLY ONCE FOR GRID POINTS
 53 --- C---- BELOW THE DYN EQUIL REGION. FOR GRID POINTS IN THE DYN EQUIL REGION
 54 --- C---- THEY ARE CALLED REPEATEDLY UNTIL THE CONJUGATE BOUNDARY IS REACHED.
```

```
 55 --- C---- TINCR=DN/DT, Q=SOURCE+SINK, PM=FUTURE CPT OF TINCR, AM=PAST CPT OF DN/
 56 ---       DO 200 JA=J,JU
 57 ---       HLL=SL(JA)-SL(JA-1)
 58 ---       HUU=SL(JA+1)-SL(JA)
 59 ---            CALL CHEMO(ION,JA,QIO,N,NSAVE,TI,THF)
 60 ---            CALL DAVE(ION,JA,ANL,ANM,ANU,PL,PM,PU,N,NSAVE)
 61 ---       DO 300 I=1,ION
 62 ---       Q(I)=Q(I)+QIO(I)
 63 ---       PM(I)=PM(I)-ANM(I)
 64 ---       TINCR(I)=TINCR(I)+PM(I)
 65 --- C.... CALCULATION OF FLUX USING THE CONTINUITY EQN. START AT LOWER BDY AND
 66 --- C.... INTEGRATE TO FAR LOWER BDY. THE BDY FLUX=THE BDY FLUX FROM MOMTM EQN
 67 ---       IF(JSJ.NE.0) GO TO 355
 68 ---       STS=1.0
 69 ---       IF(ITF.EQ.3) STS=0.0
 70 ---       FYS(I)=FYS(I)+QIO(I)-PM(I)*STS/DT
 71 ---       IF(JA.EQ.2) FYS(I)=FLL(I)
 72 ---       FY(I,JA)=BM(JA)*FYS(I)
 73 ---   355 CONTINUE
 74 --- C---- PRINT CONTINUITY PARAMS -------
 75 ---       IF(JON.NE.0)GO TO 300
 76 ---       IF(JA.LT.JLL(4).OR.J.GT.JUL(4))GO  TO 300
 77 ---       WRITE(6,666)J,JA,Q(I),TINCR(I),PM(I),ANM(I)
 78 ---   300 CONTINUE
 79 ---   200 CONTINUE
 80 --- C,,,, IF J=JB THEN CALCULATION OF THE UPPER ARC LENGTH AND BU IS DIFFERENT
 81 --- C,,,, BECAUSE THE NEXT POINT IS IN THE CONJUGATE HEMISPHERE. VEL IS CALLED
 82 --- C,,,, AGAIN TO CALC THE VELOCITY AT THE UPPER 1/2 PT
 83 ---       IF(J.NE.JB)GO TO 360
 84 ---       CALL VEL(JBS,CU,VU,FLU,N,TI,JB)
 85 ---       DELS=(SL(JBS+1)+SL(JBS)-SL(JB)-SL(JB-1))/2.0
 86 ---       BU=(BM(JBS)+BM(JBS+1))/2.0
 87 ---       GO TO 370
 88 ---   360 CALL VEL(J,CU,VU,FLU,N,TI,JB)
 89 ---       DELS=(SL(J+1)-SL(J-1))/2.0
 90 ---       BU=(BM(J)+BM(J+1))/2.0
 91 ---   370 BL=(BM(J)+BM(J-1))/2.0
 92 --- C.........
 93 --- C.... THIS SECTION SETS UP THE ERROR FUNCTIONS F(I); TINCR=DN/DT ; FGR=
 94 --- C.... FLUX CPT ; Q(I)=SOURCE+SINK. TF @ ITF ARE PARAMETERS FOR OBTAINING
 95 --- C.... VARIOUS TYPES OF SOLUTION. AT PRESENT (TF,ITF)=(3,3) IS THE ONLY
 96 --- C.... OPTION BEING USED. IT GIVES A STEADY STATE SOLUTION.
 97 ---       DO 380 I=1,ION
 98 ---       IF(ITF.EQ.3)TINCR(I)=0.0
 99 ---       FGR=-(FLU(I)/BU-FLL(I)/BL)/DELS
100 --- C.......ERROR FUNCTIONS FORMED ..........
101 ---       F(I)=(Q(I)*DT-TINCR(I))/DELS+FGR*DT
102 ---       IF(J.EQ.JB)F(I)=F(I)*40.0
103 --- C---- TINCR AND Q(I) ARE DIVIDED BY DELS TO COMPENSATE FOR THE INTER-
104 --- C---- POLATION. FGR @ Q(I) ARE MULTIPLIED BY DT INSTEAD OF DIVIDING TINCR
105 --- C---- THIS IS DONE FOR EASY COMPARISON OF PRINTED VALUES.
106 --- C---- THE VELOCITY AT THE GRID POINT V(I) IS CALCULATED AS THE AVERAGE
107 --- C---- OF THE UPPER AND LOWER VELOCITIES.
108 ---       TINCR(I)=TINCR(I)/DELS
109 ---       Q(I)=Q(I)*DT/DELS
```

```
      FGR=FGR*DT
      V(I)=.5*(VL(I)+VU(I))
      IF(JON.NE.0)GO TO 380
C.... PRINTING OF FACTORS IN THE ERROR FUNCTIONS ..........
      IF(J.LT.JLL(5).OR.J.GT.JUL(5))GO TO 380
      WRITE(6,666) J,FLU(I),FLL(I),FGR,Q(I),TINCR(I),F(I),V(I)
  666 FORMAT(2X,4H FIJ,I6,1P12E13.5)
  380 CONTINUE
      RETURN
      END
C,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+
      SUBROUTINE VEL(J,PP,V,FLUX,N,TI,JB)
C-------------------------------------------------------------------
C   THIS PROGRAM WAS WRITTEN BY EUGENE YOUNG (1978). IT CALCULATES THE
C   VELOCITIES AT THE POINT Z(J+1/2) FROM THE ION MOMENTUM EQTNS FOR
C   O+ AND H+. PARAMETERS FROM THE GRID POINT Z(J) ARE INTERPOLATED TO
C   GET THEIR VALUES  AT THE HALF POINT. THE PARAMETERS FROM ST MAURICE
C   AND SCHUNK PSS 1975 ARE TRANSFERRED FROM SUBR JP
C-------------------------------------------------------------------
        IMPLICIT REAL(A-H,L,N-Z)
       REAL Z,DT,DH,THF,EPS,MASS,MG,MAGM
      DIMENSION MASS(2),QSIGN(2),V(2),A(2),TIJ(2),PL(2),PP(3),PU(2)
     >   ,FLUX(2),ZR(2),D(2),GAMMA(2),Z12(2),B(2),Q1(2)
        DIMENSION N(4,300),TI(3,300),F(2)
      COMMON/VN/U(2,300),BG(300),BM(300),GR(2,300),R(2,300),SL(300)
      COMMON/ND/ON(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
      COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,ION,TF,ITF
      COMMON/SAV/NSAVE(2,300),TISAV(3,300),FY(2,300),UN(300),EHT(300)
      COMMON/DEQ/QION(2),DRAG(2),GP(2,300)
      COMMON/FON/JON,JLL(5),JUL(5)
      DATA A/16.0,1.0/,MASS/26.7616E-24,1.6726E-24/
      DATA QSIGN/-1.,1./,Z12/1.0,1.0/,BK/1.3807E-16/
C-------- EXPONENTIALLY INTERPOLATE AMBIENTS----------
      OA=SQRT(ON(J)*ON(J+1))
      HA=SQRT(HN(J)*HN(J+1))
      N2A=SQRT(N2N(J)*N2N(J+1))
      O2A=SQRT(O2N(J)*O2N(J+1))
      TIJ(1)=SQRT(TI(1,J)*TI(1,J+1))
      TIJ(2)=SQRT(TI(2,J)*TI(2,J+1))
      TEJ=SQRT(TI(3,J)*TI(3,J+1))
      TNJ=SQRT(TN(J)*TN(J+1))
C---------CALCULATE PARTIALLY IMPLICIT DENSITIES ,PU,PL,PM---------
      THP=1.0-THF
      NE=0.0
      NEU=0.0
      NEL=0.0
      DO 260 I=1,ION
C
          PU(I)=THP*NSAVE(I,J+1) + THF*N(I,J+1)
          PL(I)=THP*NSAVE(I,J)   + THF*N(I,J)
          PP(I)=SQRT(PU(I)*PL(I))
          NEU=NEU+PU(I)
          NEL=NEL+PL(I)
          NE=NE+PP(I)
C
```

```
165 ---    260 CONTINUE
166 --- C+++++ ADD N(ND+) AND N(D2+) TO ELECTRON DENSITY
167 ---         NEU=NEU+N(3,J+1)
168 ---         NEL=NEL+N(3,J-1)
169 ---         NE=NE+N(3,J)
170 ---         PP(3)=NE
171 --- C.... CALL JP TO OBTAIN D(I),GAMMA,AL12,ALS12 & B(I) USED TO CALC V(I)
172 ---       CALL JP(DA,HA,N2A,O2A,TIJ,TNJ,PP,D,GAMMA,AL12,ALS12,B)
173 --- C---------CALCULATE ALTITUDE DERIVATIVES----------
174 ---       DELS=(SL(J+1)-SL(J))
175 ---       DTE=(TI(3,J+1)-TI(3,J))/DELS
176 ---       DT1=(TI(1,J+1)-TI(1,J))/DELS
177 ---       DT2=(TI(2,J+1)-TI(2,J))/DELS
178 --- C,,,, TERMS IN THE MOMENTUM EQN. CONSULT ST M & S(1975) P910 EQN (19)
179 --- C,,,, THE MOMTM EQN HAS BEEN CAST IN A SLIGHTLY DIFFERENT FORM IN TERMS
180 --- C,,,, OF THE GEOPOTENTIAL. (SEE PROG NOTES). DRAG= A TERM FOR TAKING
181 --- C,,,, COLLISIONS INTO ACCOUNT IN DYN EQUIL REGION. QION=ALL THERMAL
182 --- C,,,, DIFFUSION TERMS
183 ---       DO 301 I=1,ION
184 ---       RATII=PU(I)*TI(I,J+1)/(PL(I)*TI(I,J))
185 ---       RATIE= NEU*TI(3,J+1)/(NEL*TI(3,J))
186 ---       GRA=-MASS(I)*GR(1,J+1)/(BK*TIJ(I))
187 ---       FI=A(I)*(GP(2,J+1)-GP(2,J))
188 ---       DLNI=GRA*ALOG(RATII*RATIE**Z12(I))/FI
189 ---           IK=3-I
190 ---       DRAG(I)=-(FY(I,J)*(1+B(I))/N(I,J)-FY(IK,J)/N(IK,J))/D(I)
191 ---       QION(I)=(QSIGN(I)/TIJ(I))*(GAMMA(I)*DTE
192 ---      >     -(PP(IK)/NE)*(AL12*DT1-ALS12*DT2))
193 --- C
194 ---       Q1(I)=(DLNI+GRA)+QION(I)
195 --- C.... FORM THE RHS OF THE MOMTM EQN. UN(I)=NEUTRAL WIND TERM
196 ---    729  F(I)=-D(I)*Q1(I)+UN(J)*B(I)
197 ---       IF(JON.NE.0)GO TO 301
198 --- C....PRINT ROUTINE FOR MOMTM EQN TERMS ..........
199 ---       IF(J.LT.JLL(5).OR.J.GT.JUL(5))GO TO 298
200 ---       WRITE(6,605) J,Z(J),DELS,DLNI,GRA,FI,DRAG(I),QION(I),Q1(I),
201 ---      >     RATII,RATIE
202 ---    298  IF(J.LT.JLL(3).OR.J.GT.JUL(3)) GO TO 301
203 ---       WRITE(6,605) J,Z(J),D(I),B(I),F(I),GAMMA(I),AL12,ALS12
204 ---    605  FORMAT(1X,*VEL*,I6,F9.0,1P12E12.4)
205 ---    301 CONTINUE
206 --- C--------------NOW THE VELOCITIES--------------
207 ---       DET=B(1)*B(2)+B(1)*B(2)
208 ---       V(1)=((F(1)+F(2))+B(2)*F(1))/DET
209 ---       V(2)=((F(1)+F(2))+B(1)*F(2))/DET
210 ---       FLUX(1)=V(1)*PP(1)
211 ---       FLUX(2)=V(2)*PP(2)
212 --- C.......PRINT VELOCITIES ...........
213 ---       IF(JON.NE.0)GO TO 302
214 ---       IF(J.LT.JLL(5).OR.J.GT.JUL(5))GO TO 302
215 ---       WRITE(6,606)J,V(1),V(2),DET,F(1),F(2)
216 ---    302 CONTINUE
217 ---    606  FORMAT(2X,*J=*,I4,3X,* V(1)=*,1PE11.4,* V2=*,E11.4,* DET=*
218 ---      >  ,E11.4,* F1=*,E11.4,* F2=*,E11.4)
219 ---       RETURN
```

```
      END
C,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+,!"#$%&**?+
      SUBROUTINE JP(DA,HA,N2A,O2A,TIJ,TNJ,PP,D,GAMMA,AL12,ALS12,B)
C-----------------------------------------------------------------------
C--- THIS ROUTINE PROVIDES THE PARAMETERS ON P910 OF ST MAURICE
C--- @ SCHUNK PSS 1975.  EQUATIONS 20,21,22,23,24,25,26 AND THE
C--- RATIO OF NEUTRAL/ION COLL FREQS-B.  IT USES AVERAGE DENSITIES
C--- FROM THE INTERPOLATION ROUTINES TERD AND TERS
C-----------------------------------------------------------------------
        IMPLICIT REAL(A-H,N-Z)
      DIMENSION A(4),AMR(2,4),TIST(2,2),TIJ(2),NU(2,2),AMUN(2,4)
     >   ,TINT(2,4),NUX(2),NX(2,4),D1(2,2),D4(2,2),NUP(2,2),POL(6)
     >   ,CR(6),AR(6),BR(6),D(2),GAMMA(2),B(2),Z(2),PP(3)
      DATA BK/1.3807E-16/,AMU/1.6726E-24/
C--------ION NEUTRAL PARAMETERS FROM BANKS AND KOCKARTS 1973 P244
C----  TABLE 9.13 POLARIZABILITY (POL), MULTIPLICATIVE (CR)
C----    A TERM (AR) AND B TERM (BR) P219
      DATA  POL/.79,.667,1.76,1.59,.21,1.1/
      DATA  CR/4.8E-13,1.9E-12,3.7E-13
     >   ,3.4E-13,9.7E-13,5.2E-13/
      DATA   AR/10.6,14.4,14.3,10.6,11.6,10.4/
      DATA  BR/.67,1.17,.96,.76,1.05,.64/
      DATA A/16.0,1.0,28.0,32.0/,Z/1.0,1.0/,S2/1.41421/
C     THE FOLLOWING CODE COMES FROM J-P*S SUBROUTINE FREQ
C--------------REDUCED MASS AND TEMPERATURE-----------
      DO 11 I=1,2
      DO 11 J=1,4
      AMR(I,J)=A(I)*A(J)/(A(I)+A(J))
      TINT(I,J)=(A(J)*TIJ(I)+A(I)*TNJ)/(A(I)+A(J))
      IF(J.GT.2)GO TO 11
      TIST(I,J)=(A(J)*TIJ(I)+A(I)*TIJ(J))/(A(I)+A(J))
   11 CONTINUE
C---- ION-NEUTRAL COLLISION FREQUENCIES -----------
      GM=2.0*TINT(1,1)
      NX(1,1)=CR(1)*DA*SQRT(GM)*(AR(1)-BR(1)*ALOG10(GM))
      GM=2.0*TINT(2,2)
      NX(2,2)=CR(2)*HA*SQRT(GM)*(AR(2)-BR(2)*ALOG10(GM))
      NX(1,2)=5.0E-11*HA*SQRT(TINT(1,2))
      NX(2,1)=4.4E-11*DA*SQRT(TINT(2,1))
      DO 111 I=1,2
      NUX(I)=0.0
      NX(I,3)=2.6E-9*SQRT(POL(3)/AMR(I,3))*(N2A)
      NX(I,4)=2.6E-9*SQRT(POL(4)/AMR(I,4))*(O2A)
      DO 111 J=1,4
      NUX(I)=NUX(I)+NX(I,J)
  111 CONTINUE
C... ION-ION COLLISIONS FROM ST.M.@S P920 DIVIDED BY 2.15 SEE
C... SCHUNK @WALKER PSS 1970 P1319 (1324)
      DO 113 I=1,2
      DO 113 J=1,2
      NU(I,J)=1.2726*Z(I)*Z(I)*Z(J)*Z(J)*PP(J)*
     >  SQRT(AMR(I,J))/(A(I)*2.15*TIST(I,J)**1.5)
      AA=(A(I)+A(J))**2
      D4(I,J)=(1.2*A(J)*A(J)-1.5*A(I)*A(J))/AA
  113 CONTINUE
```

```
TING                                                                    06/27,

 275 --- C.... NUP=NU' (FROM ST M @ S) ..........
 276 ---       K=3
 277 ---       DO 20 I=1,2
 278 ---       K=K-1
 279 ---       DO 20 J=1,2
 280 ---       NUP(I,J)=1.25*NU(I,J)*(D4(I,J)+1.5*AMR(I,J)/A(I)
 281 ---      A*TIJ(I)/TIST(I,J))
 282 ---       IF(I.EQ.J) NUP(I,J)=NU(I,J)+1.25*NU(I,K)*(D1(I,K)
 283 ---      A+1.5*AMR(I,K)/A(I)*TIJ(I)/TIST(I,K))
 284 --- 20    CONTINUE
 285 --- C  ..............................
 286 --- C   THE FOLLOWING CODE IS FROM J-P*S SUBROUTINE TRANSCD
 287 --- C  ..............................
 288 ---       N12=PP(1)/PP(2)
 289 ---       M12=A(1)/A(2)
 290 ---       X=NU(1,2)/NUP(1,1)*(1.-NUP(2,1)/NUP(2,2))
 291 ---       X=X/(1.-NUP(1,2)*NUP(2,1)/(NUP(1,1)*NUP(2,2)))
 292 ---       AL12=X*TIJ(1)/TIST(1,2)*15./8.*(N12+1.)/(M12+1.)
 293 ---       ALS12=AL12*M12*M12*TIJ(2)/TIJ(1)*
 294 ---      A(NUP(1,1)-NUP(1,2))/(NUP(2,2)-NUP(2,1))
 295 ---       DELTA=3./(5.*(N12+1.)*(M12+1.))*(AL12+N12*M12*ALS12)
 296 ---       CT=BK/(AMU*(1.-DELTA))
 297 ---       DO 40 I=1,2
 298 ---       J=3-I
 299 --- C---- DIFFUSION COEFF D(I) AND RATIO OF COLL FREQS B(I) -------
 300 ---       D(I)=CT*TIJ(I)/(A(I)*NU(I,J))
 301 ---       B(I)=  NUX(I)/NU(I,J)
 302 --- 40    CONTINUE
 303 ---       GAMMA(1)=15./8.*S2*(PP(3)*Z(1)*Z(1)-Z(1)*(PP(1)*Z(1)*Z(1)
 304 ---      >    +PP(2)*Z(2)*Z(2)))/(13.*S2/8.*(PP(1)*Z(1)*Z(1)
 305 ---      >   +PP(2)*Z(2)*Z(2))+PP(3))
 306 ---       GAMMA(2)=15.*S2/8.*(Z(2)*(PP(1)*Z(1)*Z(1)
 307 ---      >    +PP(2)*Z(2)*Z(2))-PP(3)*Z(2)*Z(2))/
 308 ---      >   (13.*S2/8.*(PP(1)*Z(1)*Z(1)+PP(2)*Z(2)*Z(2))+PP(3))
 309 ---       RETURN
 310 ---       END

ATION DATASET IS ABOUT TO BE WRITTEN

GIVEN,  RF DONE
ATION DATASET HAS NOW BEEN WRITTEN

-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    43/100 SECONDS IS ELAPSED TI
```

```
DIT,D=PLIB,DN=RSLPS,DL
  1 --- C.........(RSLPS)...........
  2 ---         SUBROUTINE LOOPS(S,NEQ,N,TI,ITAU,JB,SEC,DNDT)
  3 --- C-------------------------------------------------------------
  4 --- C    SUBROUTINE LOOPS IS THE MAIN SEQUENCING CONTROL PROGRAM. IT CALLS SUB-
  5 --- C    PROGS DFIJ & TFIJ TO OBTAIN THE ERROR FUNCTIONSFIJ FOR THE DENSITY
  6 --- C    AND TEMPERATURE DIFFERENTIAL EQUATIONS.IT SETS UP THE JACOBIAN MATRIX IN
  7 --- C    SUBR JMATRX AND SOLVES FOR THE INCREMENTS USING BDSLV. BEFORE EACH TIME
  8 --- C    THE SOLUTIONS TO THE EQUATIONS ARE GUESSED USING A CUBIC PREDICTOR. THE
  9 --- C    INCREMENTS FROM THE SOLVER ARE ALSO TESTED TO ENSURE NON NEGATIVE
 10 --- C    DENSITIES (MODIFIED STEEPEST DESCENT). FINALLY, THE RESULTS ARE PRINTED
 11 --- C    BY CALLING SUBR PRINT NOTE THAT IF THE # OF GRID POINTS=300 THE
 12 --- C    FOLLOWING ARRAYS HAVE THESE DIMENSIONS-- DELTA,RHS,SHR ALL 1200
 13 --- C    ,WORK(7500). THE ARRAYS ARE USED IN JMATRX AND BDSLV TO SOLVE THE
 14 --- C    JACOBIAN MATRIX. DCRQ=ARRAY FOR SAVING INDICATORS OF NONCONVERGENCE.
 15 --- C    NP/TEP=PREDICTED DENS/TEMP. NOTE THAT BECAUSE OF THE DYN EQUIL REGION
 16 --- C    THE SIZE OF THE MATRIX VARIES FROM THE TEMPERATURE TO THE DENSITY EQNS
 17 --- C    JBS,MIT,IEQ AND JB ARE ALL USED TO ACCOMPLISH THIS.
 18 --- C-------------------------------------------------------------
 19 ---           IMPLICIT REAL (A-H,L,N-Z)
 20 ---         REAL  Z,DT,DH,THF,EPS,TF
 21 ---           INTEGER ION,NEQ,KS,K,NFLAG
 22 ---         DIMENSION S(NEQ,1),DELTA(1200),RHS(1200),WORK(7500),TEP(300)
 23 ---        >    ,SHR(1200),DCRQ(2,300),NP(4,300),V(2)
 24 ---         DIMENSION N(4,300),TI(3,300),F(2),DNDT(2,300),DTDT(3,300),VS(2)
 25 ---         COMMON/VN/U(2,300),BG(300),BM(300),GR(2,300),R(2,300),SL(300)
 26 ---         COMMON/ND/DN(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
 27 ---         COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,ION,TF,ITF
 28 ---         COMMON/SAV/NSAVE(2,300),TISAV(3,300),FY(2,300),UN(300),EHT(300)
 29 ---         COMMON/LPS/EPSN,DC,IMOD,I1,I2,IPRIN,IPMX,IPP,ISKP
 30 ---         COMMON/LQ/L(2),Q(2)
 31 ---          COMMON/DEQ/QION(2),DRAG(2),GP(2,300)
 32 ---          DATA  JTI,ICALL,DTSAV,ITSAV
 33 ---        >    /   0  ,  3  ,  1.  ,-10000/
 34 ---            NT=0
 35 ---            JTI=JTI+1
 36 ---            JBS=JMAX-JB+1
 37 ---            MIT=2*JB-1
 38 ---            IEQ=2*(MIT-2)
 39 ---            JBP1=JB+1
 40 --- C    DNDT,DTDT ARE DERIVATIVES USED IN  PREDICTING FUTURE VALUES OF DENSITY A
 41 --- C    TEMP, DNDTS & DTDTS ARE SAVED DERIVS FROM PREVIOUS TIME STEP ALSOUSEDIN
 42 --- C    PREDICTION BELOW 200KM LOCAL LOSS/GAIN EQUIL IS USED TO PREDICT DENSITY
 43 --- C    & TEMP THROUGH A CALL TO HDEQ. NP IS USED TO STORE THE PREDICTED
 44 --- C    VALUE SO THAT ITS ACCURACY CAN BE USED TO IMPROVE THE NORMAL QUADRATIC P
 45 --- C    ICTION TO A CUBIC. DENSITIES ARE SAVED IN NSAVE BEFORE PREDICTION OCCURS
 46 --- C.... NT= TOTAL TUBE CONTENT. CALL HDEQ TO INITIALIZE GP
 47 ---         CALL HDEQ(1,2,JMAX,N,TI,R,SL)
 48 ---           DNDTS=0
 49 ---           DTDTS=0
 50 --- 99      DO 41 I=1,3
 51 ---         DO 41 J=1,JMAX
 52 --- C.... NOW PREDICT TEMP USING QUADRATIC PREDICTER.............
 53 ---           IF(JTI.NE.1) DTDTS=DTDT(I,J)
 54 ---           IF(JTI.NE.1)   DTDT(I,J)=(TI(I,J)-TISAV(I,J))/DTSAV
```

```
 55 ---           IF(JTI.EQ.1) DTDT(I,J)=0
 56 --- C---- AT LOW ALTS USE HOEQ TO PREDICT FUTURE DENSITIES AND TEMPS
 57 --- C---- BUT FIRST SAVE OLD TEMPS
 58 ---          TKP=TISAV(I,J)
 59 ---          TISAV(I,J)=TI(I,J)
 60 ---          IF((Z(J).LT.200).AND.(I.EQ.3))
 61 ---       >    CALL HOEQ(J,JMAX,DT,THF,DCR,NP,TI,NSAVE)
 62 ---          IF(I.EQ.3) TEP(J)=TI(3,J)
 63 ---          TI(I,J)=((DTDT(I,J)-DTDTS)*DT/DTSAV+DTDT(I,J))*DT+TISAV(I,J)
 64 ---          IF(TI(I,J).LE.0.0) TI(I,J)=TISAV(I,J)
 65 --- C-------- DON'T USE PREDICTOR AT A TURNING POINT ----
 66 ---          IF(DTDT(I,J)*DTDTS.LT.0.0) TI(I,J)=(TISAV(I,J)+TKP)/2
 67 ---          IF(ITF.EQ.3) TI(I,J)=TISAV(I,J)
 68 ---          IF((Z(J).GT.200).AND.(I.EQ.3)) TEP(J)=TI(3,J)
 69 ---          IF(I.GT.ION) GO TO 412
 70 --- C.... NOW PREDICT DENSITIES USING CUBIC PREDICTER.....
 71 ---           IF(JTI.NE.1) DNDTS=DNDT(I,J)
 72 ---           IF(JTI.NE.1)   DNDT(I,J)=(N(I,J)-NSAVE(I,J))/DTSAV
 73 ---           IF(JTI.LT.2) DNDTS=DNDT(I,J)
 74 ---          NKP=NSAVE(I,J)
 75 ---           NSAVE(I,J)=N(I,J)
 76 ---           IF(I.EQ.1) NT=NT+BM(1)*(N(2,J)+N(1,J))*DH/(BG(J)*BM(J))
 77 ---          IF(JTI.EQ.1) N(I,J)=DNDT(I,J)*DT+N(I,J)
 78 ---          IF(JTI.GT.1) N(I,J)=((DNDT(I,J)-DNDTS)*DT/DTSAV+DNDT(I,J))
 79 ---       > *DT+N(I,J)
 80 ---          IF(N(I,J).LE.0.0) N(I,J)=NSAVE(I,J)
 81 --- C-------- DON'T USE PREDICTOR AT A TURNING POINT ----
 82 ---          IF(DNDT(I,J)*DNDTS.LT.0.0) N(I,J)=(NSAVE(I,J)+NKP)/2
 83 ---          IF(ITF.EQ.3) N(I,J)=NSAVE(I,J)
 84 ---  411   NP(I,J)=N(I,J)
 85 ---        GO TO 41
 86 --- C,,,, THIS SECTION BLENDS THE ORIGINAL PREDICTION INTO THE PHOTO-CHEM
 87 --- C,,,, EQUILIBRIUM PREDICTION TO PROVIDE A SMOOTH JOIN AT 200KM
 88 ---  412    IF(Z(J).GT.200) GO TO 41
 89 ---         A=5*EXP(15*(Z(J)-200)/200)
 90 ---         N(1,J)=(N(1,J)*A+NP(1,J))/(1+A)
 91 ---         N(2,J)=(N(2,J)*A+NP(2,J))/(1+A)
 92 ---         TI(3,J)=(TI(3,J)*A+TEP(J))/(1+A)
 93 ---         NP(1,J)=N(1,J)
 94 ---         NP(2,J)=N(2,J)
 95 ---         TEP(J)=TI(3,J)
 96 ---  41      CONTINUE
 97 --- C.........
 98 --- C......... END PREDICTOR ...........
 99 ---          WRITE(6,114) NT
100 --- C... ALTERNATING CALL LOOP; IF IFIJ=1 OBTAIN TEMPERATURES
101 --- C... IF IFIJ=2 OBTAIN DENSITIES
102 ---         DO 226 IFIJ=11,12
103 --- C**** ITER LOOP;ON EACH ITERATION THE JACOBIAN IS FORMED AND SOLVED FOR
104 --- C**** THE INCS OF DENSITY OR TEMP. THE DENSITY OR TEMP ARE UPDATED.
105 --- C**** IF THE DENSITIES AT SUCCESSIVE ITERATES ARE SUFFICIENTLY
106 --- C**** CLOSE THE SOLUTION HAS BEEN FOUND.
107 ---        DO 220 ITER=1,30
108 ---          IVC=0
109 --- C.......BOUNDARY CONDITIONS ON DENSITY AND TEMPERATURE
```

```
110 ---           CALL HDEQ(1,JMAX,DT,THF,DCR,N,TI,NSAVE)
111 ---           CALL HDEQ(JMAX,JMAX,DT,THF,DCR,N,TI,NSAVE)
112 --- C+++++ COMPUTE FIJ   PROFILE AND *S* MATRIX WHICH BDSLV USES TO
113 --- C+++++ SOLVE THE LINEAR SYSTEM ((S))*(DELTA)= (FIJ)
114 --- C.... DO LOOP FOR CALLING FIJ ....
115 ---         DO 145 J=2,JMAX1
116 ---          IF((IPR.EQ.2).AND.(J.EQ.MIT)) GO TO 147
117 ---          KR=2*(J-2)
118 ---          JA=J
119 ---          IF((IPR.EQ.2).AND.(J.GT.JB)) JA=J+JBS-JB
120 ---          IF(IFIJ.EQ.1)  CALL TFIJ(JA,0,IPR,N,TI,F,JB,JBS,V)
121 ---          IF(IFIJ.EQ.2)  CALL DFIJ(JA,0,IPR,N,TI,F,JB,JBS,V)
122 ---          RHS(KR+1)=F(1)
123 ---          RHS(KR+2)=F(2)
124 ---          IF(IPR.NE.2) GO TO 145
125 --- C... CONVERGENCE CRITERIA ON VELOCITY IF REQUIRED
126 ---          DO 143 I=1,2
127 ---          VS(I)=U(I,JA)
128 ---          IF(V(I).EQ.0.0) GO TO 143
129 ---          IF(ABS((V(I)-VS(I))/V(I)).GT.1.E-4) IVV=IVV+1
130 --- 143      U(I,JA)=V(I)
131 ---    145 CONTINUE
132 --- 147       CONTINUE
133 --- C.... IJM IS A SWITCH FOR CALLING JMATRIX OR NOT ....
134 ---          IJM=0
135 ---          IF((ITER/ICALL)*ICALL.EQ.ITER) IJM=1
136 ---          IF(ITER.LE.3) IJM=1
137 ---          IF((IPR.EQ.3).AND.(ITER.EQ.2)) IJM=1
138 ---          IBD=NEQ
139 ---          IF(DCRT.NE.1) IJM=1
140 ---          IF(DCR.NE.1) IJM=1
141 ---          IF(IPR.EQ.2) IBD=IEQ
142 ---          IF(IJM.EQ.1) CALL JMATRX(S,RHS,IBD,IPR,DT,JMAX,N,TI,F,JB,JBS
143 ---        >    ,MIT)
144 --- C++++ INVERT THE JACOBIAN MATRIX *S* AND RECORD ERROR COMMENTS
145 --- C++++      FROM THE INVERSION ROUTINE *BDSLV* ON FILE *IDCB*
146 ---         CALL BDSLV(IBD,3,S,0,RHS,DELTA,WORK,NFLAG)
147 ---         IF(NFLAG.EQ.0)GO TO 55
148 ---         WRITE(6,215) ITER,M,NFLAG
149 ---         STOP 3
150 --- 55        IDIV=0
151 ---           DCR=1
152 ---           DCRP=1.0
153 ---           DCRT=1.0
154 ---           IF(IPR.NE.3) GO TO 135
155 --- C....... TEST 'TI' TO ENSURE TI>0 (MOD STEEP DESCENT) ......
156 ---          DO 137 I=1,IPR
157 ---          DO 137 J=2,JMAX1
158 ---          ION=3
159 ---          IF(I.EQ.IPR) ION=2
160 ---          DINC=DELTA(2*J-ION)
161 ---          IF(ABS(DINC/TI(I,J)).GT.EPSN) DCRP=DC*ABS(TI(I,J)/DINC)
162 ---          IF(DCRP.LT.DCRT) DCRT=DCRP
163 ---   137    CONTINUE
164 ---   135    IF(IPR.NE.2) GO TO 144
```

```
165 --- C,,,,,, TEST 'N' TO ENSURE N>0 (MOD STEEPEST DESCENT) ,,,,,,
166 ---         DO  142 I=1,IPR
167 ---         DO 142 J=2,JMAX1
168 ---         IDN=3
169 ---         IF(I.EQ.IPR) IDN=2
170 ---          IJ=J
171 ---         IF(J.GT.JB) IJ=J+JB-JBS
172 ---         IF((J.GT.JB).AND.(J.LE.JBS))   GO TO 142
173 ---         DINC=DELTA(2*IJ-IDN)
174 ---         IF(ABS(DINC/N(I,J)).GT.EPSN) DCRP=DC*ABS(N(I,J)/DINC)
175 ---         IF(DCRP.LT.DCR) DCR=DCRP
176 ---         IF(ITER.EQ.1) DCRQ(I,J)=DCRP
177 --- 142      CONTINUE
178 --- C... ADD ITERATIVE INCREMENT TO THE ARRAY 'N' OR 'TI' AND TEST FOR
179 --- C... CONVERGENCE WHEN IDIV=0
180 --- 144     DO  42 I=1,IPR
181 ---         IDN=3
182 ---         IF(I.EQ.IPR) IDN=2
183 ---         DO 42 J=2,JMAX1
184 ---         DINC=DELTA(2*J-IDN)
185 ---         IF(IPR.EQ.2) GO TO 59
186 ---         TI(I,J)=TI(I,J)-DINC*DCRT
187 ---         IF(ABS(DINC/TI(I,J)).GT.EPS)   IDIV=IDIV+1
188 ---         GO TO 42
189 --- 59        CONTINUE
190 ---          IJ=J
191 ---         IF(J.GT.JB) IJ=J+JB-JBS
192 ---         IF((J.GT.JB).AND.(J.LE.JBS))   GO TO 42
193 ---         DINC=DELTA(2*IJ-IDN)
194 ---         N(I,J)=N(I,J)-DINC*DCR
195 ---         IF(ABS(DINC/N(I,J)).GT.1.E-9)  IDIV=IDIV+1
196 --- 42       CONTINUE
197 --- C.....
198 ---          IF(IPR.NE.2) GO TO 47
199 ---          IF(JB.EQ.JBS)GO TO 47
200 --- C.... FILL IN DYNAMIC EQUILIBRIUM REGION
201 ---        DO 43 J=JBP1,JBS
202 ---        CALL HUDEQ(J,J-1,JMAX,N,TI,R,SL)
203 ---    43 CONTINUE
204 --- 47       IF((ITER/3)*3.EQ.ITER) WRITE(6,113)
205 ---         WRITE(6,112) ITER,ITAU,IDIV,IVC,DCR,DCRT
206 --- C...IF DCR HAS NOT GONE TO 1 BY THE TENTH ITERATION, GIVE UP.
207 ---         IF((DCR.NE.1).AND.(ITER.EQ.10)) GO TO 230
208 ---         IF((DCRT.NE.1).AND.(ITER.EQ.10)) GO TO 230
209 --- C*** TEST TO SEE IF CONVERGENCE HAS OCCURED.
210 ---          IF(IDIV+IVC.EQ.0)  GO TO 224
211 ---   220 CONTINUE
212 ---        GO TO 230
213 --- 224      IF(IFIJ.EQ.I2) GO TO 222
214 ---  226        CONTINUE
215 ---  222        CONTINUE
216 --- C*** SAVE THE TIME STEP FOR THE PREDICTOR AND CALL THE PRINT SUBROUTINE
217 ---          DTSAV=DT
218 ---         IF(ITF.EQ.3) ITSAV=-10000
219 ---         IF(ITAU-ITSAV.LT.ISKP) RETURN
```

```
INC                                                                    06/27/

220 ---          ITSAV=ITAU
221 ---          CALL PRINT(IPMX,IPP,Z,JMAX,N,TI,ITAU,ITER,6,U,FY,NP)
222 ---          RETURN
223 --- C**********
224 --- 230        CONTINUE
225 ---          CALL PRINT(IPMX,1,Z,JMAX,N,TI,ITAU,ITER,6,DCRQ,FY,NP)
226 ---          WRITE(6,115)
227 ---           STOP
228 ---    215 FORMAT(*  ITERATION =*,I5,*  TIME STEP =*,I5,
229 ---       A/,*  RETURN FROM BDSLV =*,I5)
230 ---    111 FORMAT(1H ,I4,1P9E12.2)
231 ---    112 FORMAT(1H ,4I7,1P9E12.2)
232 --- 113      FORMAT(/4X,*ITER*,4X,*ITAU*,4X,*IDIV*,4X,*IVC*,4X,*DCR*)
233 --- 114      FORMAT(/*TOTAL TUBE CONTENT = *,1PE13.5)
234 --- 115      FORMAT(*NON-CONVERGENCE STOP*)
235 ---         END

TION DATASET IS ABOUT TO BE WRITTEN

IVEN,  RF DONE
TION DATASET HAS NOW BEEN WRITTEN

+-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    28/100 SECONDS IS ELAPSED TI
RAY1
LL BE CHARGED AT BACKGROUND RATE
IMIT,T=6000S,PT=10S,PR=300,PU=0,DD80=0
SSIGN,P3POSYM=2,9,16
SSIGN,P8D3PO=1,9,16
SSIGN,SCR=6
SSIGN,SCR=7
LIB,1,BN,MA,DS=600,RP,SP
LIB,2,BN,MD,DS=600,RP,SP
ORTRAN,S=PLIB,SN=RSMN
DIT,S=PLIB,SN=RSMN,D=3521,ON=CRAY1

  0       FILE GENERATED ON 06/27/79  AT    0.81

TION DATASET IS ABOUT TO BE WRITTEN
TION DATASET HAS NOW BEEN WRITTEN

+-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    36/100 SECONDS IS ELAPSED TI
ORTRAN,S=PLIB,SN=GTS3S
DIT,S=PLIB,SN=GTS3S,D=3521,ON=CRAY1

  0       FILE GENERATED ON 05/23/79  AT  11.98

TION DATASET IS ABOUT TO BE WRITTEN
TION DATASET HAS NOW BEEN WRITTEN

+-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    29/100 SECONDS IS ELAPSED TI
ORTRAN,S=PLIB,SN=RSPR
DIT,S=PLIB,SN=RSPR,D=3521,ON=CRAY1

  0       FILE GENERATED ON 05/23/79  AT  11.98
```

```
EDIT,D=PLIB,DN=PSXIN,DL
   1 ---        PROGRAM PSXIN
   2 --- C.....(PSXIN).....
   3 --- C... THIS MAIN PROGRAM CREATES A FILE OF DUMMY INITIAL CONDITIONS
   4 --- C... TO BE USED IN TEMPERATURE AND DENSITY PROGRAMS
   5 --- C... THE PROGRAM CALLS AMBS FOR NEUTRAL PARAMS,FIELD FOR FIELD PARAMS.
   6 --- C... AND HYD FOR H+ DENSITIES;FOR DETAILS OF THE VARIOUS PARAMETERS
   7 --- C... CONSULT MAIN PROGRAM. ANOX IS CALLED FOR O+ DENSITIES.
   8 ---        IMPLICIT REAL(N)
   9 ---        REAL MDK
  10 ---          INTEGER NION,NDY,NY
  11 ---          DIMENSION  D(7),T(2),GL(300),XS(300),RN(2,300),RTI(3,300)
  12 ---       >    ,DS(159),FD(9),SZA(300)
  13 ---        COMMON/VN/N(2,300),U(2,300),BG(300),BM(300),GR(300),TI(3,300)
  14 ---        COMMON/ND/DN(300),HN(300),N2N(300),O2N(300),PHIDN(300),TN(300)
  15 ---        COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,NION
  16 ---        COMMON/NUSAVE /NSAVE(2,300),TISAV(3,300),FY(2,300),UN(300),AUG
  17 ---         COMMON/FDN/JDN,IPR,IKEEP,ITAU
  18 --- C... MOST OF THE DATA PARAMETERS CAN BE FOUND IN THE MAIN PROGRAM.
  19 --- C... TEE= INITIAL ELECTRON TEMPERATURE,ZCHEM=HEIGHT OF CHEMICAL
  20 --- C... EQUILIBRIUM(UNUSED).ZCO=HEIGHT FOR O+ DIFUSSIVE EQUILIBRIUM
  21 ---          DATA PCO,SCAL,ZO ,ISW,IRIT,JPRIN, TEE,MMAX, EPS ,THF,PTS
  22 ---       >     / 2.2,  5 ,120, 2 ,  1 , 125 ,4000, 9  ,1.E-3, 1 ,120/
  23 ---          DATA  IDAY ,  SEC ,BLON,F107A,F107,AP,DEC,ETRAN,ZCHEM
  24 ---       >     / 74111 , 68232, 60 , 75  , 71 ,31,12 ,43100,500 /
  25 ---          DATA  RE , DT ,IPIC,VO,   MDK  , ZCH , ZCO ,HTOP,AVMU,EM
  26 ---       >    /  6370,1200,  4 , 0,1.2114E-8,1200,1.E6,5000,.577,1.5 /
  27 ---          DATA  RE , DT ,IPIC,VO,   MDK  , ZCH , ZCO ,HTOP,AVMU,EM
  28 ---       >    /  6370,1200,  4 , 0,1.2114E-8,1200,1.E6,5000,.577,1.5 /
  29 ---          DATA   P1  , P2 ,  P3 ,   P4  ,  P5  ,    P6
  30 ---       >      / 1051,17.7,-2746,-5.1E-4,6.1E-6,-3.4E-8  /
  31 --- C++++   PCO=L SHELL, PTS=# OF PTS FOR 1/2 FIELD LINE
  32 --- C++++ SCAL=SCALING FACTOR BETWEEN 1 AND 5
  33 ---        WRITE(6,951)
  34 ---   951 FORMAT(*PCO,  PTS  ,SCAL=0 READ(1),ZO *)
  35 ---        WRITE(6,952)ISW,PCO,PTS,SCAL,ZO
  36 ---   952 FORMAT(I2,4F10.3)
  37 ---          IF(SCAL.NE.0.) GO TO 41
  38 ---        REWIND 1
  39 ---        READ(1)JMAX,PCO,ZO,SCAL,IDAY,SEC,ETRAN,F107,F107A,AP,BLON,
  40 ---       1 DEC,((N(I,J),J=1,JMAX),I=1,2),((TI(I,J),J=1,JMAX),I=1,3)
  41 ---       1   ,((U(I,J),J=1,JMAX),I=1,2)
  42 --- 41         CONTINUE
  43 --- C***TIME  STEP : END NEAR LINE 135
  44 ---        DO 221 JTI=1,MMAX
  45 --- C-----
  46 ---        IF(JTI.EQ.1) GO TO 661
  47 --- C... INFORMATION REQUIRED FOR MAIN PROGRAMS IS WRITTEN TO TAPE
  48 ---        WRITE(2)JMAX,PCO,ZO,SCAL,IDAY,SEC,ETRAN,F107,F107A,AP,BLON,
  49 ---       1 DEC,((N(I,J),J=1,JMAX),I=1,2),((TI(I,J),J=1,JMAX),I=1,3)
  50 ---       1   ,((U(I,J),J=1,JMAX),I=1,2)
  51 ---        STOP
  52 --- 661      CONTINUE
  53 --- C... XY=STARTING VALUE FOR NEWTON ITERATION IN SOLUTION OF STEADY STATE
  54 --- C... ION TEMPERATURE;DH=X-DISTANCE BETWEEN POINTS; DINTO IS USED IN
```

```
 55 --- C... DIFFUSIVE EQUIL PROFILE FOR O+ ; ITEST IS USED FOR CONTINUOUS ENERGY
 56 --- C...   LOSS TEST;
 57 ---           XY=1000.0
 58 ---        DH=1./(PTS-1.)
 59 ---          IPTS=PTS
 60 ---          XS(1)=0.
 61 ---          NHE=0.
 62 ---          DINTO=0.
 63 ---          JMAX=2*IPTS-1
 64 ---          ITEST=0
 65 --- C-----ESTABLISH PTS. ON FIELD LINE BY CALL TO FIELD
 66 --- C.... R=RADIUS TO FIELD PT ; Z=ALT ; SDIP=SIN(DIP ANG) ; GL=GEOM LAT
 67 --- C.... GR=GRAVITY ; BM IS PROP TO MAG FIELD STRENGTH ; BG=VARIABLE
 68 --- C.... COORD FACTOR ; DS=STEP SIZE(KM) ; XS=ARC LENGTH FROM EQUAT (KM
 69 --- C
 70 ---        DO 950 J=1   ,IPTS
 71 ---          CALL FIELD(J,JMAX,PCO,RE,ZO,SCAL,X,FD)
 72 ---          Z(J)=FD(1)
 73 ---          SDIP=FD(2)
 74 ---          CDIP=FD(3)
 75 ---          GL(J)=FD(4)
 76 ---          GR(J)=FD(5)
 77 ---          BM(J)=FD(6)
 78 ---          BG(J)=FD(7)
 79 ---          XS(J)=FD(8)
 80 ---          IF(J.GT.1) DS(J)=XS(J-1)-XS(J)
 81 ---          IF(J.EQ.IPTS) GO TO 1000
 82 --- C.... CONJUGATE FIELD PARAMETERS .....
 83 ---        JU=JMAX+1-J
 84 ---        GL(JU)=-GL(J)
 85 ---        GR(JU)=-GR(J)
 86 ---        Z(JU)=Z(J)
 87 ---        BM(JU)=BM(J)
 88 ---        BG(JU)=BG(J)
 89 --- 950         CONTINUE
 90 --- 1000        CONTINUE
 91 ---          DO 17 I=1,JMAX
 92 --- C---------TRANSFORM MAGNETIC TO GEOGRAPHIC COORDINATES---
 93 --- C-------- AND CALCULATE SZ ANGLE AT EACH FIELD LINE POINT BY CALL TO AMBS---
 94 ---          ZZJ=Z(I)
 95 ---          GLAT=GL(I)
 96 --- C*****
 97 ---         CALL AMBS(I,IDAY,SEC,ZZJ,GLAT,DEC,ETRAN,F107,
 98 ---       >  F107A,AP,BLON,D,T,CHI)
 99 --- C*****
100 ---        ON(I)=D(2)
101 ---        HN(I)=D(7)
102 ---        N2N(I)=D(3)
103 ---        O2N(I)=D(4)
104 ---        TN(I)=T(2)
105 ---          SZA(I)=CHI
106 ---        U(1,I)=0
107 ---        U(2,I)=0
108 --- 17        CONTINUE
109 --- C....
```

```
110 --- C,,,,,  END NEUT; SET INIT. TE AND ION DENSITY PROFILES, OX(I)
111 --- C... ANALYTICAL O+ DENSITY IS OBTAINED FROM SUBR ANOX
112 ---          DO 317 I=1,IPTS
113 ---          IF(ISW.EQ.3)  GO TO 21
114 ---        ZZJ=Z(I)
115 ---        CALL ANOX(I,ZZJ,OX)
116 ---        IF(Z(I).LT.750.) N(2,I)=1.25*EXP(0.01198*Z(I))
117 ---        IF(Z(I).GE.750.) N(2,I)=1249*EXP(2.773E-3*Z(I))
118 ---        IF(Z(I).GE.1000.0) N(2,I)=28284*EXP(-3.5E-4*Z(I))
119 --- 19           CONTINUE
120 ---          IF(ISW.EQ.2) TI(3,I)=TEE
121 --- C... THE ION TEMPERATURE IS OBTAINED FROM THE UNIFORM ELECTRON TEMP
122 --- C... USING THE STEADY STATE CONDITION HEATING=COOLING:
123 --- C... THIS IS DONE BY A NEWTON ITERATIVE PROCEDURE;USUALLY TE IS THEN
124 --- C... CALCULATED AS THE AVERAGE OF THE UNIFORM TE AND TI
125 --- 21         CONTINUE
126 ---          G1=(N(2,I)+OX/16.)*(OX+N(2,I))/TI(3,I)**1.5
127 ---          G2=2.7586E-10*OX*ON(I)+1.8E-9*N(2,I)*HN(I)
128 ---          G3=(OX*(8.67*N2N(I)+3.678*NHE)+7.225*N(2,I)*NHE)*1.0E-9
129 ---          G4=(4.729E-10*OX*HN(I)+5.255E-10*N(2,I)*ON(I))*SQRT(TN(I))
130 ---          FEX=G1*(TI(3,I)-XY)-G2*((SQRT(XY+TN(I)))*(XY-TN(I)))
131 ---       1    -(G3+G4)*(XY-TN(I))
132 ---          DEX=-(G1+G3+G4)-.5*G2*(3*XY+TN(I))/SQRT(XY+TN(I))
133 ---          XY=XY-FEX/DEX
134 ---          IF(ABS(FEX/(DEX*XY)).GT.1.0E-4)  GO TO 21
135 ---           TI(1,I)=XY
136 ---           TI(2,I)=XY
137 ---          IF(ISW.EQ.2) TI(3,I)=(TI(3,I)+TI(2,I))/2.
138 --- C,,,,  ALTER INITIAL PROFILES TO DIFFUSIVE EQUILIBRIUM IF NECESSARY
139 ---          N(1,I)=OX
140 ---         IF(Z(I).LT.ZCO) GO TO 29
141 ---          NE=N(1,I)+N(2,I)
142 ---         DINTO=-16.*MOK*GR(I)*DH/(BG(I)*(TI(2,I)+N(1,I)*TI(3,I)/NE))
143 ---         N(1,I)=N(1,I-1)*EXP(DINTO)
144 ---     29  CONTINUE
145 --- C.... TEST TO SEE IF CONTINUOUS ENERGY LOSSES ARE TOO LARGE ....
146 --- C.... IF SO CHANGE SCAL OR PTS TO REDUCE STEP SIZE
147 --- C.... AVMU=<COSO> ; EM = LOW ENERGY OF P.E.
148 --- 117        CONTINUE
149 --- C          ET=8.618E-5*TI(3,I)
150 --- C          ZNE=N(1,I)+N(2,I)
151 --- C       TEST=((3.37E-12*ZNE**0.97)/(EM**0.94))*((EM-ET)/
152 --- C      1(EM-(0.53*ET)))**2.36
153 --- C       TEST=TEST*DH/(AVMU*BG(I))
154 --- C          IF(Z(I).GT.1000) DE=DE+TEST
155 --- C       IF (TEST .LT. 1.0) GO TO 317
156 --- C          ITEST=ITEST+1
157 --- C       IF(ITEST.EQ.1) WRITE(6,955)
158 --- C 955 FORMAT(*CONTINUOUS ENERGY LOSS*)
159 --- C          WRITE(6,663) I,Z(I),TEST
160 --- 317        CONTINUE
161 ---        CALL HYD
162 --- C****  GENERATE THE CONJUGATE HEMISPHERE  *****
163 ---        DO 110 J=1,IPTS-1
164 ---        JU=JMAX+1-J
```

```
165 ---          IF(ISW.EQ.1)  GO TO 110
166 ---         N(1,JU)=N(1,J)
167 ---         N(2,JU)=N(2,J)
168 ---         TI(1,JU)=TI(1,J)
169 ---         TI(2,JU)=TI(2,J)
170 --- C,,,, ANALYTICAL CALC OF TE FOR LOW ALTITUDES. SPLICES INTO EXIST
171 --- C,,,, ING TE AT 600 KM. SEE BRACE AND THEIS GRL(1978) P275
172 ---          IF(Z(J).LT.600)
173 ---        > ATE=P1+(P2*Z(J)+P3)*EXP(P4*Z(J)+P5*N(1,J)+P6*Z(J)*N(1,J))
174 ---          AV=EXP((Z(J)-600)/50)
175 ---          BV=EXP((400-Z(J))/50)
176 ---          IF(Z(J).LT.400) AV=0
177 ---          IF(Z(J).LT.600) TI(3,J)=(TI(3,J)*AV+ATE*BV)/(AV+BV)
178 ---         TI(3,JU)=TI(3,J)
179 ---  110   CONTINUE
180 --- C..... PRINT SPECIFICATIONS .....
181 ---           IF(IRIT.EQ.0) GO TO 221
182 ---           WRITE(6,1020)
183 ---         DO 106 J=1,JPRIN
184 ---         WRITE(6,111)J,Z(J),GL(J),BM(J),BG(J),GR(J),N(1,J),N(2,J),SZA(J)
185 ---         IF((J/9)*9.EQ.J) WRITE(6,1020)
186 ---    106 CONTINUE
187 ---         DO 105 J=1,JPRIN
188 ---           IF(J.EQ.1) WRITE(6,112)
189 ---           IF((J/9)*9.EQ.J) WRITE(6,112)
190 ---           IF((J*1)*1.EQ.J)WRITE(6,113)   J,Z(J),ON(J),HN(J),N2N(J),O2N(J)
191 ---       1     ,TN(J),TI(3,J),TI(1,J)
192 ---   105     CONTINUE
193 --- C......
194 ---    221 CONTINUE
195 --- 616           FORMAT(1H ,'   ISW=1 RETAIN TEMPS CALC NEW DENS'/
196 ---       1     '   ISW=2 CALC COMPLETELY NEW PROFILES'/
197 ---       1     '   ISW=3 RETAIN TE BUT CALC NEW TI'/
198 ---       1      '   ISW=4 FIND NEW H+ WITH OLD O+'/)
199 ---    663 FORMAT(I5,2F10.3)
200 --- 665           FORMAT(I7,12F10.2)
201 --- 1020        FORMAT(3X,'J',8X,'Z',8X,'GL',8X,'BM',8X,'BG',8X,'GR'
202 ---       1      ,9X,'OX',9X,'HYD')
203 --- 112       FORMAT(3X,'J',8X,'Z',9X,'ON',9X,'HN',9X,'N2N',9X,'O2N'
204 ---       1      ,9X,'TN',9X,'TE',9X,'TI')
205 ---    111 FORMAT(I4,F10.4,F10.6,1X,E10.4,1X,E9.4,F10.4,4E12.5)
206 ---    113 FORMAT(I4,F10.4,1X,E12.6,1X,E12.6,1X,E12.4,1X,E12.6,1X
207 ---       >          3F10.4)
208 ---         END
209 --- C....ANOX..ANOX..ANOX..ANOX..ANOX..ANOX..ANOX..ANOX
210 ---         SUBROUTINE ANOX(J,Z,OX)
211 --- C-------------------------------------------------------
212 --- C    THIS PROGRAM FITS ANALYTICAL FUNCTIONS TO ANY EXPERIMENTAL
213 --- C    O+ PROFILE THE F2 PEAK IS A PARBOLIC MAX AT PT 3. THE VALLEY IS
214 --- C    AT A PARBOLIC MIN AT PT 1. THE F2 PEAX DENSITY IS A PARABOLIC MAX
215 --- C    THE FIRST 4 POINTS SHOULD BE EQUIDISTANT IN ALTITUDE.
216 --- C    ABOVE THE ALTITUDE CORRESPONDING TO PT 4 EXPONENTIALS ARE USED
217 --- C    THERE CAN BE ANY NUMBER OF THESE SECTIONS.
218 --- C    READ THE ALTS AND DENSITIES FROM FILE 5.
219 --- C-------------------------------------------------------
```

```
220 ---        DIMENSION ZP(10),A(10),B(10),DEN(10)
221 ---        DATA IPTS/7/,(ZP(I),I=1,7)/120,200,230,260,300,500,1000/
222 ---        DATA (DEN(I),I=1,7)/1E4,3.5E5,4.4E5,3.5E5,2.1E5,3.2E4,5E3/
223 ---        IF(J.GT.1) GO TO 5
224 ---        B(1)=DEN(2)
225 ---        A(1)=-(DEN(1)-DEN(2))/(ZP(2)-ZP(1))**2
226 ---        B(2)=DEN(4)
227 ---        A(2)=-(DEN(3)-DEN(4))/(ZP(4)-ZP(3))**2
228 ---        IPTS2=IPTS-2
229 ---        DO 6 I=3,IPTS2
230 ---        B(I)=ALOG(DEN(I+2)/DEN(I+1))/(ZP(I+2)-ZP(I+1))
231 ---        A(I)=DEN(I+1)/EXP(B(I)*ZP(I+1))
232 ---      6 CONTINUE
233 ---      5 IS=0
234 ---        IPTS1=IPTS-1
235 ---        DO 10 I=1,IPTS1
236 ---        IF(Z.GE.ZP(I)) IS=I
237 ---     10 CONTINUE
238 ---        IF(IS.GT.3) GO TO 15
239 ---        IF(Z.LE.ZP(2)) DX=A(1)*(Z-ZP(2))*(Z-2*ZP(1)+ZP(2))+B(1)
240 ---        IF(Z.GE.ZP(2)) DX=A(2)*(Z-ZP(4))*(Z-2*ZP(3)+ZP(4))+B(2)
241 ---        IF(IS.LT.4) RETURN
242 ---     15  IS=IS-1
243 ---     20 DX=A(IS)*EXP(B(IS)*Z)
244 --- C      RETURN
245 ---        END
246 --- C....FIELD..FIELD..FIELD..FIELD..FIELD..FIELD..FIELD
247 ---          SUBROUTINE FIELD(J,JMAX,PCO,RE,ZO,SCAL,X,FD)
248 --- C-------------------------------------------------------
249 --- C    THIS PROGRAM DETERMINES THE GRID POINT SPACING GIVEN JUST JMAX=
250 --- C    # OF GRID POINTS, PCO=LSHELL,ZO=LOWER BOUNDARY, SCAL=SCALLING
251 --- C    FACTOR;; X=COLATITUTE, THE FIELD PARAMETERS ARE TRANSFERRED
252 --- C    THROUGH FD(I);; RO=EQUATORIAL RADIUS TO FLUX TUBE,INITIAL VALUES
253 --- C    FOR X AND R ARE SET. DH=DISTANCE BETWEEN POINTS IN THE X COORDINATE
254 --- C-------------------------------------------------------
255 ---          DIMENSION FD(9)
256 ---          IF(J.NE.1) GO TO 950
257 ---           RO=RE*PCO
258 ---          R=RO
259 ---          X=.5
260 ---          PTS=(JMAX+1)/2
261 ---          IPTS=PTS
262 --- C'''' SCALK IS A SCALING FACTOR TO ENSURE THAT THE X-COORD.
263 --- C'''' RANGES FROM 0 TO 1
264 ---         RAT=(RE+ZO)/RE
265 ---          QMAX=(SQRT(1.-RAT/PCO))/(RAT**2)
266 ---          SCALK=1./SINH(SCAL*(QMAX))
267 ---          DH=1/(PTS-1)
268 ---          RAT=(RE/RO)**2
269 --- C-----ESTABLISH PTS. ON FIELD LINE
270 --- C.... R=RADIUS TO FIELD PT ; FD1=ALT ; FD2=SIN(DIP ANG) ; FD4=GEOM LAT
271 --- C.... FD5=GRAVITY ; FD6 IS PROP TO MAG FIELD STRENGTH ; FD7=VARIABLE
272 --- C.... COORD FACTOR ; DS=STEP SIZE(KM) ; FD8=ARC LENGTH FROM EQUAT (KM
273 --- C
274 ---   950 DX=1.-(J-1)/(PTS-1.0)
```

```
275 ---           IF(J.EQ.IPTS)     GO TO 4
276 --- C... Q=THE DIPOLE COORD DETERMINED FROM SINH(KQ)=DX
277 ---            SCX=DX/SCALK
278 ---           Q=ALOG(SCX+SQRT(SCX**2+1))/SCAL
279 --- 3         SHX=SIN(X)
280 ---          CHX=COS(X)
281 --- C... THE NEXT 6 LINES ARE A NEWTON SOLVER FOR THE EQUATION F(X)=0
282 --- C... THIS DETERMINES THE COLATITUTE X
283 ---           FEX=(RO**2)*(SHX**4)-(RE**2)*CHX/Q
284 ---           DEX=(RO**2)*4.*(SHX**3)*CHX+(RE**2)*SHX/Q
285 ---           X=X-FEX/DEX
286 ---           IF(ABS(FEX/(DEX*X)).GT.1.0E-6) GO TO 3
287 ---           GO TO 5
288 --- 4         X=1.570796327
289 ---           SHX=1.
290 ---           CHX=0.
291 --- C ----LAT. RADIUS. DIP
292 --- 5         R=RO*SHX**2
293 ---           FD(1)=R-RE
294 ---           SQTH=SQRT(3.*(CHX**2)+1.)
295 ---           FD(2)=2.*CHX/SQTH
296 ---           FD(3)=SHX/SQTH
297 ---           FD(4)=1.570796327-X
298 ---           FD(5)=FD(2)*3.98E+10/((RE+FD(1))**2)
299 ---           FD(6)=1.0E-5*(RE**2)*SQTH /(R**3)
300 ---           FD(7)=SCAL*COSH(SCAL*Q)*FD(6)*SCALK
301 ---           DS=1.0E-5*DH/FD(7)
302 ---           XX=ALOG(1.732*CHX+SQTH)
303 ---           FD(8)=RO*.28868*(XX+SINH(XX)*COSH(XX))
304 ---           RETURN
305 ---         END
306 --- C.....................................
307 ---           SUBROUTINE AMBS(J,IDAY,SEC,ZZJ,GL,DEC,ETRAN,F107,
308 ---         >  F107A,AP,BLON,D,T,SZA)
309 --- C-------------------------------------------------------------
310 --- C    THIS PROGRAM EVALUATES THE SOLAR ZENITH ANGLE,DETERMINES THE
311 --- C    THE DAY AND TIME AND CALLS A.E. HEDINS MSIS MODEL(GTS3S)TO GET
312 --- C    THE NEUTRAL DENSITIES AND TEMPERATURE
313 --- C-------------------------------------------------------------
314 ---           DATA  PLAT , PLON , HEC , KTRAN ,  PI   ,ICNT
315 ---        1  /   1.375 , 1.222 , 0.0 , 43200 , 3.14159, 0 /
316 --- C---------CORRECT TIME FOR FULL DAYS AND YEARS---
317 --- 999     IF(J.NE.1) GO TO 1000
318 ---         NY=INT(IDAY/1000)
319 ---         NDY=MOD(IDAY,1000)
320 ---         IF(MOD(NY,4).EQ.0)LY=366
321 ---         IF(MOD(NY,4).NE.0)LY=365
322 ---         SX=AMOD(SEC,86400.)
323 ---         NDY=NDY+INT(SEC/86400)
324 ---         NY=NY+INT(NDY/LY)
325 ---         ID=1000*NY+MOD(NDY,LY)
326 --- 1000        CONTINUE
327 --- C---------TRANSFORM MAGNETIC TO GEOGRAPHIC COORDINATES---
328 --- C--------- AND CALCULATE SZ  ANGLE AT EACH FIELD LINE POINT---
329 ---         BLOR=PLON*PI/180.0
```

```
330 ---         XM=COS(GL)*COS(BLOR-PLON)
331 ---         YM=COS(GL)*SIN(BLOR-PLON)
332 ---         ZM=SIN(GL)
333 ---         XG=XM*SIN(PLAT)+ZM*COS(PLAT)
334 ---         YG=YM
335 ---         ZG=-XM*COS(PLAT)+ZM*SIN(PLAT)
336 ---         GLAT=ASIN(ZG)
337 ---         GLATD=GLAT*180./PI
338 ---         GLON=(PLON+ATAN2(YG,XG))*180./PI
339 ---         SAT=(SX-ETRAN+43200.0)/3600.-GLON/15.0
340 ---         HH=(SAT-12.)*15.*PI/180.
341 ---         S7A=ACOS(COS(GLAT)*COS(DEC)*COS(HH)
342 ---        >    +SIN(GLAT)*SIN(DEC))
343 --- C......CALL GTS3S FOR NEUTRAL PARAMS.........
344 ---         CALL GTS3S(ID,SX,ZZJ,GLATD,GLON,SAT,F107A,F107,AP,48,D,T)
345 ---    665 FORMAT(I7,9F7.1)
346 ---         RETURN
347 ---         END
348 --- C.....HYD...HYD...HYD...HYD...HYD...HYD...HYD...HYD
349 ---            SUBROUTINE HYD
350 --- C,,,, HYD PRODUCES A GOOD INITIAL PROFILE(H+0 FROM A UNIFORM
351 --- C,,,, PROFILE HDENS=CONSTANT , USING A SHOOTING METHOD SEE
352 --- C,,,, P RICHARDS PHD THESIS 1978. ALSO BAILEY ET AL PSS 1978 P753.
353 --- C... ARRAYS A,B,C,D,E ARE STORAGE ARRAYS,ARRAY T IS A STORE FOR N(H+)
354 --- C... GRAD=ALTITUDE BELOW WHICH H+ GRADIENT IS TESTED.TMAX/TMIN=MAX/MIN
355 --- C... DENSITY ALLOWED.
356 ---         IMPLICIT REAL(N)
357 ---           DIMENSION A(150),B(150),C(150),D(150),E(150),T(150)
358 ---         REAL  Z,DT,DH,THF,EPS,GRAD,TMAX,TMIN
359 ---           INTEGER NION,NEQ
360 ---         COMMON/VN/N(2,300),U(2,300),BG(300),BM(300),GR(300),TI(3,300)
361 ---         COMMON/ND/CN(300),HN(300),N2N(300),O2N(300),PHION(300),TN(300)
362 ---         COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,NION
363 ---         COMMON/NUSAVE /NSAVE(2,300),TISAV(3,300),FY(2,300),UN(300),AUG
364 ---          COMMON/FON/JON,IPR,IKEEP,ITAU
365 --- C... VSTART=EQUATIORAL DENSITY FOR SEARCHING PROCEDURE ;G25=SWITCH TO
366 --- C... GUIDE THE SEARCH; VMORE VLESS ARE 'USED IN BINARY SPITTING.
367 --- C... HDENS=UNIFORM INITIAL H+ DENSITY. IGO IS A SWITCH FOR VARIOUS
368 --- C... MODES OF OPERATION. ZLBDY IS THE LOWER BOUNDARY FOR H+ SOLUTION.
369 ---            COMMON/A1/VSTART,G25,VMORE,VLESS
370 ---           DATA     RE   , IFREQ,     BK      , DELTA , HDENS , IGO , ZLBDY
371 ---        1     /   6374 ,  10  ,  8.2522E+7,  10     ,2.5E+3,  5 ,  460 /
372 ---           DATA   IRIT , IT  , ZPRIN ,GRAD , TMAX , TMIN,IPRIN
373 ---        1     /   0    ,   0 , 1000  , 500 , 1.E5 , 10 , 0 /
374 ---           DH2=2*DH
375 ---           DT=1200
376 ---           IEQ=(JMAX+1)/2
377 ---           TKEEP=N(2,IEQ)
378 ---           TI(3,IEQ+1)=TI(3,IEQ-1)
379 ---           TI(2,IEQ+1)=TI(2,IEQ-1)
380 ---           N(1,IEQ+1)=N(1,IEQ-1)
381 ---           N(2,20)=HDENS
382 --- C
383 --- 16       DO 17  I=1,IEQ
384 ---           UN(I)=0
```

```
385 ---           IF(Z(I).LT.ZLBDY) I400=I
386 ---           FY(1,I)=0
387 --- 17          CONTINUE
388 --- C... THE DO 20 LOOP EVALUATES THE VARIOUS TERMS IN THE MOMENTUM
389 --- C... EQUATION, THAT IS GRAVITY,THERMAL DIFFUSION COLLISION FREQUENCIES)
390 ---           DO 20  I=I400-1,IEQ
391 ---           AL=0
392 ---           DTE=-(TI(3,I+1)-TI(3,I-1))/DH2
393 ---           DTI=-(TI(2,I+1)-TI(2,I-1))/DH2
394 ---           TSQT=SQRT(TI(2,I))
395 ---           V20=6.957E-11*TSQT*ON(I)+5.4E-9*HN(I)
396 ---           V12=.075/(TSQT*TI(2,I))
397 ---           V21=1.2/(TSQT*TI(2,I))
398 ---           IF(IGO.EQ.5) N(2,I)=HDENS
399 --- C.... THERMAL DIFFUSION COEFF AL ....
400 ---           IF(Z(I).LT.450) GO TO 23
401 ---           IF(Z(I).GT.2500) GO TO 23
402 ---           EN=N(1,I)/N(2,I)
403 ---           B1=-0.149481
404 ---           B2=1.250519+0.583095/EN
405 ---           B3=2.750533+2.332381*EN
406 ---           AL=0.0882353*(EN+1)*(16*(B3/EN-B1)+(B1/EN-B2))
407 ---          1   /(B1**2-B2*B3)
408 --- 23        CONTINUE
409 ---          NSAVE(2,I)=N(2,I)
410 ---           C1=BG(I)*BK*((DTI+DTE)
411 ---          1      -TI(3,I)*(N(1,I+1)-N(1,I-1))/(DH2*(N(1,I)+N(2,I))))
412 ---           C2=-GR(I)+V20*UN(I)
413 ---           C3=-BK*N(1,I)*AL*DTI*BG(I)/(N(1,I)+N(2,I))
414 ---           C(I)=C1+C2+C3
415 ---           D(I)=V21*N(1,I)+V20
416 ---           E(I)=U(2,I)
417 --- C         IF(ABS(U(2,I)).GT.YMAX) WRITE(6,201) U(2,I)
418 --- C         WRITE(6,205)   I,Z(I),C1,C2,C3,C(I),D(I),DTI,DTE
419 --- C     1     ,N(1,I),BG(I)
420 --- 20        CONTINUE
421 ---                IF(IGO.EQ.5)  IGO=1
422 --- C... THE DO 24 LOOP EVALUATES THE PRODUCTION AND LOSS TERMS IN THE
423 --- C... CONTINUITY EQUATION.
424 ---              DO 24 I=I400-1,IEQ
425 ---            BETA=2.3E-11*SQRT(TI(2,I))*ON(I)
426 ---            PRODH=2.5E-11*SQRT(TN(I))*HN(I)*N(1,I)
427 ---            IF(I.EQ.I400) VCHEM=PRODH/BETA
428 ---              A(I)=(PRODH+IGO*N(2,I)/DT)/(BG(I)*BM(I))
429 ---              B(I)=-(BETA+IGO/DT)/(BG(I)*BM(I))
430 ---              IF(IRIT.NE.2)  GO TO 24
431 ---              IF(I.EQ.1)  WRITE(6,211)
432 ---              WRITE(6,205)  I,Z(I),A(I),B(I),C(I),D(I)
433 --- 24            CONTINUE
434 --- C... VSTART IS THE EQUATIORAL DENSITY TO START THE SEARCHING PROCEDURE
435 --- C... DELTA IS THE INCREMENT TO ADD ON TO VSTART IF SEARCH IS UNSUCCESSFUL
436 --- C... UNTIL BINARY SPLITTING STARTS IJK=COUNTER TO DETERMINE NON-CONVERGENCE
437 ---             IF(N(2,IEQ).EQ.N(2,20)) IGO=0
438 ---              IF(IGO.EQ.0) TKEEP=N(2,IEQ)
439 ---              VSTART=2*N(2,IEQ)-TKEEP
```

```
440 ---            DELTA=0.1*ABS(TKEEP-N(2,IEQ))
441 ---            TKEEP=N(2,IEQ)
442 ---            IJK=0
443 ---            RAT=.05
444 ---
445 --- C::::::IF VSTART IS NO GOOD COME BACK TO HERE & START AGAIN
446 --- 26        N(2,IEQ)=VSTART
447 ---          T(IEQ)=VSTART
448 ---           IJK=IJK+1
449 ---           IF(IJK.EQ.4) DELTA=10
450 ---           IF(IGO.EQ.0)  DELTA=.3*N(2,IEQ)
451 ---           IF(IGO.EQ.0) WRITE(6,204) IJK,VSTART,VLESS,VMORE,VCHEM
452 ---           IF(IJK.EQ.50)   STOP
453 ---           PHIT=0
454 ---           I=0
455 ---           J=IEQ
456 --- C;;;;;;COME BACK TO HERE TO FIND NEXT VALUE ALONG FIELD LINE
457 --- C
458 --- 25       J=J-1
459 ---          I=J+1
460 ---          DELPHI=DH*(A(I)+B(I)*N(2,I)) /(RE*1.0E+5)
461 ---          PHIT=PHIT+DELPHI
462 ---          FY(2,I)=(PHIT)*BM(I)*RE*1.0E+5
463 ---          U(2,J)=FY(2,I)/N(2,I)
464 ---          F1=BG(I)*BK*(TI(2,I)+N(2,I)*TI(3,I)/(N(1,I)+N(2,I)))
465 ---          F2=-0.5*(C(I)+C(I-1))/F1- 0*FY(2,I)*D(I)/(F1*N(2,I))
466 ---          F3=-FY(2,I)*D(I)/(F1*N(2,I))
467 ---          F4=0
468 ---          F5=0
469 ---          F6=0
470 ---          IF(IT.LT.4) GO TO 251
471 ---          F4=FY(1,I)*16*V12/F1
472 ---          F5=-(U(2,J)-E(J))/(DT*F1)
473 ---          F6=-E(I)*(E(J)-E(I))*BG(I)/(F1*DH)
474 --- 251      ALOG=(F2+F3+F5+F6)*DH
475 ---          YMAX=SQRT(F1/BG(I))
476 ---          N(2,J)=N(2,I)*EXP(ALOG)
477 ---          T(J)=N(2,J)
478 ---          IF(U(2,J).LT.-YMAX)  N(2,J)=5.0E+5
479 ---          IF(U(2,J).GT.YMAX)   N(2,J)=2.0
480 --- C.... GO INTO PRINT ROUTINE .....
481 ---          IF(IPRIN.NE.0) GO TO 51
482 ---          IF(IJK.LT.200) GO TO 50
483 --- 51       CONTINUE
484 --- C         GO TO 50
485 ---          IF(Z(I).LT.ZPRIN) GO TO  39
486 ---          IF(I.EQ.IEQ) GO TO 39
487 ---          IF((I/IFREQ)*IFREQ.NE.I) GO TO 50
488 --- 39       WRITE(6,205) J,Z(J),NSAVE(2,J),N(2,J),F1,F2,F3,FY(2,I)
489 --- C
490 --- 50      CALL BIN(J,T,TMIN,TMAX,GRAD,DELTA,I400,VCHEM,RAT)
491 ---         IF(G25-1)   26,25,40
492 ---
493 --- 40       CONTINUE
494 ---          DO 42 I=1,I400
```

```
      N(2,I)=1.13*HN(I)*N(1,I)*SQRT(TN(I))/(ON(I)*SQRT(TI(2,I)))
   42   CONTINUE
      RETURN
  201     FORMAT(3X,'U(2,I)=',1PE11.4)
  204   FORMAT(1H ,I4,3X,1P6E13.5)
  205     FORMAT(I4,F9.2,1P11D11.2)
  211   FORMAT(1H ,3X,'I',8X,'Z(I)',6X,'A(I)',7X,'B(I)',7X,'C(I)',
     1    8X,'D(I)')
        STOP
        END
C**BIN****BIN****BIN****BIN****BIN****BIN****BIN****BIN****BIN****BIN**
      SUBROUTINE BIN(J,TEST,YMIN,YMAX,GRAD,DELTA,ITMAX,VCHEM,RAT)
C .............<B.A> ..........
C... THIS IS THE BINARY SPLITTING ROUTINE TEST=DENSITY TO BE TESTED.
      IMPLICIT REAL(L,N)
     REAL  Z,DT,DH,THF,EPS
      INTEGER NION,NEQ
      REAL YMIN,YMAX,GRAD
     COMMON/ALT/Z(300),JMAX,JMAX1,DT,DH,THF,ITER,EPS,NION
      COMMON/A1/VSTART,G25,VMORE,VLESS
      DIMENSION  TEST(300)
      DATA   ISEQ , VMORE , VLESS
     1     /   0    ,   0  ,    0 /
      G25=0
C... TEST TO SEE IF LOWER BOUNDARY HAS BEEN REACHED, IF SO, TEST DENSITY
C... AGAINST CHEMICAL EQUILIBRIUM.
      IF(J.GT.ITMAX)   GO TO 51
      VDIFF=TEST(J)-VCHEM
      IF(ABS(VDIFF/TEST(J)).LT.RAT)   GO TO 42
      IF(VDIFF)  51,53,53
C... IF SUCCESSFUL SOLUTION INITIALIZE AND RETURN G25=2 TO INDICATE
C... SUCCESS TO MAIN PROGRAM.
   42   VMORE=0
      VLESS=0
      ISEQ=0
      G25=2
      RETURN
C... TEST TO SEE IF DMIN<DENSITY<DMAX
   51   IF(J.EQ.ITMAX)  GO TO 511
      IF(TEST(J).GT.YMIN)   GO TO 52
  511   IF(ISEQ.EQ.1)  GO TO 56
C... DENSITY IS TOO SMALL, INCREASE, OR BINARY SPLIT.
      ILESS=1
      VLESS=VSTART
      IF(VMORE.NE.0)    GO TO 55
      VSTART=VSTART+DELTA
      RETURN
C... TEST TO SEE ALTITUDE IS LOW ENOUGH FOR GRADIENT TO MATTER.
C... IF NOT TEST TO SEE IF DENSITY IS TOO BIG.
   52   IF(Z(J).LE.GRAD) GO TO 53
      IF(TEST(J).LT.YMAX)   GO TO 58
      IF(ISEQ.EQ.1)  GO  TO  57
C... DENSITY IS TOO BIG REDUCE VSTART OR BINARY SPLIT.
      IMORE=1
      VMORE=VSTART
```

```
 550 ---            IF(VLESS.NE.0)  GO TO  55
 551 ---            VSTART=VSTART-DELTA
 552 ---            RETURN
 553 --- C
 554 --- 53        IF(J.EQ.ITMAX)   GO TO 531
 555 ---            IF(TEST(J)-TEST(J+1).LT.0)      GO TO 58
 556 --- 531       IF(ISEQ.EQ.1)   GO TO 57
 557 ---            IMORE=1
 558 ---            VMORE=VSTART
 559 ---            IF(VLESS.NE.0)  GO TO 55
 560 ---            VSTART=VSTART-DELTA
 561 ---            RETURN
 562 --- C
 563 --- 55        ISEQ=1
 564 ---            VSTART=(VLESS+VMORE)*.5
 565 ---            RETURN
 566 --- C
 567 --- 56        VLESS=VSTART
 568 ---            VSTART=(VMORE+VSTART)*.5
 569 ---            RETURN
 570 --- C
 571 --- 57        VMORE=VSTART
 572 ---            VSTART=(VLESS+VSTART)*.5
 573 ---            RETURN
 574 --- C... IF DENSITY IS OKAY, RETURN G25=1 TO PROCEED TO NEXT POINT.
 575 --- 58         G25=1
 576 ---           RETURN
 577 ---           END

ATION DATASET IS ABOUT TO BE WRITTEN

GIVEN,  RF DONE
ATION DATASET HAS NOW BEEN WRITTEN

-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    57/100 SECONDS IS ELAPSED TI
CRAY1
ILL BE CHARGED AT BACKGROUND RATE
LIMIT,T=10S,PT=10S,PR=50,PU=C,DD80=0
ASSIGN,P1D2P2=2,9,16
TLIB,2,BN,MD,DS=600,RP,SP
FORTRAN,S=PLIB,SN=PSXIN
EDIT,S=PLIB,SN=PSXIN,D=3521,DN=CRAY1

   0        FILE GENERATED ON 06/27/79  AT    8.23

ATION DATASET IS ABOUT TO BE WRITTEN
ATION DATASET HAS NOW BEEN WRITTEN

-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    39/100 SECONDS IS ELAPSED TI
FORTRAN,S=PLIB,SN=GTS3S
EDIT,S=PLIB,SN=GTS3S,D=3521,DN=CRAY1

   0        FILE GENERATED ON 05/23/79  AT   11.98
```

```
- *EDIT,U=PLIB,DN=RSJAC,LL
      1 --- C..............(RSJAC).....................
      2 ---         SUBROUTINE JMATRX(S,RHS,NEQ,IPR,DT,JMAX,N,TI,F,JB,JBS,MIT)
      3 --- C--------------PARABOLIC DENSITY EQUATIONS VERSION---------------
      4 --- C     JMATRX EVALUATES DELF/DELT USING STEPHANSONS METHOD.
      5 --- C           ITS ELEMENTS ARE GIVEN BY:
      6 --- C                      DEL( FIJ(IF,JF) )
      7 --- C           S(L,M) = --------------------
      8 --- C                      DEL( TI(IV,JV) )
      9 --- C     WHERE N IS THE DENSITY OF ION IV AT ALTITUDE JV. L AND M
     10 --- C      ARE COMPUTED FROM JF...IV. THE ARRAY RHS CONTAINS VALUES OF
     11 --- C           FIJ SAVED FROM PREVIOUS COMPUTATION
     12 --- C-------------------------------------------------------
     13 ---           IMPLICIT REAL (A-H,N-Z)
     14 ---          REAL DT
     15 ---           INTEGER NION,NEQ
     16 ---         DIMENSION RHS(1200),S(NEQ,17),N(2,300),TI(3,300),F(2),V(2)
     17 --- C
     18 ---           WRITE(6,33)
     19 --- 33      FORMAT(*ENTER JMATRX*)
     20 ---          DATA  FRACT , FRACD
     21 ---        >  /   1.E-4   , 1.E-7  /
     22 ---           JMAT=JMAX
     23 ---           IF(IPR.EQ.2) JMAT=MIT
     24 ---           JMAT1=JMAT-1
     25 --- C
     26 ---         DO 180 JZS=1,NEQ
     27 ---         DO 190 KZS=1,7
     28 ---         S(JZS,KZS)=0.0
     29 ---    190  CONTINUE
     30 ---    180  CONTINUE
     31 --- C
     32 ---         DO 80 JF=2,JMAT1
     33 ---         J1=MAXO(2,JF-1)
     34 ---         J2=MINO(JF+1,JMAT1)
     35 ---         DO 90 JV=J1,J2
     36 ---         DO 130 IV=1,2
     37 --- C
     38 ---         L=2*(JF-2)
     39 ---         IF(JF.LE.3)M=2*(JV-2)+IV
     40 ---         IF(JF.GT.3)M=2*(JV-JF)+IV+4
     41 ---         KRV=2*(JV-2)+IV
     42 --- C
     43 --- C... H FOR TI CALCULATIONS ..........
     44 ---           IF(IPR.EQ.2) GO TO 68
     45 ---         H=FRACT*TI(IV+1,JV)/(1.0+0*DT)
     46 ---         TI(IV+1,JV)=TI(IV+1,JV)+H
     47 ---         CALL TFIJ(JF,1,IPR,N,TI,F,JB,JBS,V)
     48 ---         TI(IV+1,JV)=TI(IV+1,JV)-H
     49 ---          GO TO 70
     50 --- 68        CONTINUE
     51 ---         IF(JV.LE.JB)   H=FRACD*N(IV,JV)/(1.0+0*DT)
     52 ---         IF(JV.GT.JB)   H=FRACD*N(IV,JV+JBS-JB)/(1.0+0*DT)
     53 ---         IF(JV.LE.JB)   N(IV,JV)=N(IV,JV)+H
     54 ---         IF(JV.EQ.JB)   NIK=N(1,JBS)
```

```
      IF(JV.EQ.JB)  NIC=N(2,JBS)
      IF(JV.GT.JB)  N(IV,JV+JBS-JB)=N(IV,JV+JBS-JB)+H
C
      IF(JF.LE.JB)  CALL DFIJ(JF,1,IPR,N,TI,F,JB,JBS,V)
      IF(JF.GT.JB)  CALL DFIJ(JF+JBS-JB,1,IPR,N,TI,F,JB,JBS,V)
      IF(JV.LE.JB)  N(IV,JV)=N(IV,JV)-H
      IF(JV.EQ.JB)  N(1,JBS)=NIK
      IF(JV.EQ.JB)  N(2,JBS)=NIC
      IF(JV.GT.JB)  N(IV,JV+JBS-JB)=N(IV,JV+JBS-JB)-H
   70   CONTINUE
      FU1=F(1)
      FU2=F(2)
      FL1=RHS(L+1)
      FL2=RHS(L+2)
C
      IF(JF.LE.3)  S(L+1,M)=(FU1-FL1)/H
      IF(JF.LE.3)  S(L+2,M)=(FU2-FL2)/H
      IF(JF.GT.3)  S(L+1,M-1)=(FU1-FL1)/H
      IF(JF.GT.3)  S(L+2,M-2)=(FU2-FL2)/H
C
  130 CONTINUE
   90 CONTINUE
   80 CONTINUE
      RETURN
      END
C*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,
      SUBROUTINE BDSLV(N,M,S,KS,B,X,WORK,NFLAG)
      IMPLICIT REAL(A-H,O-Z)
      DIMENSION S(N,1),WORK(N,1)
      NFLAG=0
      I1=2*M+2
      I2=I1+1
      I3=I2+1
C
  600 FORMAT(' BDSLV ')
      WRITE(6,600)
C
      CALL BNDX(N,M,S,KS,B,X,WORK,WORK(1,I1),WORK(1,I2),WORK(1,I3)
     A,NFLAG)
      RETURN
      END
C*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,?!*$#%@,
      SUBROUTINE BNDX(N,M,A,KS,B,X,S,SCALE,INDEX,MLL,NFLAG)
      IMPLICIT REAL(A-H,O-Z)
      DIMENSION A(N,1),B(1),X(1),SCALE(1),INDEX(1),MUL(M,1)
     A,S(N,1)
      DIMENSION MES5(5),MES2(3),MES3(4)
      REAL MUL,INDEX
      IF(M.NE.0) GO TO 103
      DO 102 I=1,N
      IF(A(I,1)) 101,118,101
  101 X(I)=B(I)/A(I,1)
  102 CONTINUE
      RETURN
  103 IF(KS.NE.0) GO TO 126
```

```
110 ---          IBW=2*M+1
111 ---          IF(M.GT.N-1) GO TO 113
112 ---          DO 105 J=1,IBW
113 ---          DO 104 I=1,N
114 ---          S(I,J)=A(I,J)
115 --- 104     CONTINUE
116 --- 105     CONTINUE
117 --- C
118 --- C
119 ---          NP1=N+1
120 ---          MP1=M+1
121 ---          DO 108 I=1,M
122 ---          MP1PI=MP1+I
123 ---          IF(MP1PI-IBW) 106,106,109
124 --- 106     NP1MI=NP1-I
125 ---          DO 107 J=MP1PI,IBW
126 ---          S(I,J)=0.
127 ---          S(NP1MI,J)=0.
128 --- 107     CONTINUE
129 --- 108     CONTINUE
130 --- 109     DO 115 I=1,N
131 ---          BIG=0.
132 ---          DO 111 J=1,IBW
133 ---          IF(BIG-ABS(S(I,J))) 110,111,111
134 --- 110     BIG=ABS(S(I,J))
135 --- 111     CONTINUE
136 ---          IF(BIG) 114,112,114
137 ---    112 WRITE(6,919)I
138 --- 919     FORMAT('     IN BDSLV, ROW',I3,' IS ZERO IN INPUT MATRIX')
139 ---          NFLAG=2
140 ---          RETURN
141 --- 113     WRITE(6,918)
142 --- 918     FORMAT('     IN BDSLV &&&&&&& BANDWIDTH IS TOO LARGE')
143 ---          NFLAG=3
144 ---          RETURN
145 --- 114     SCALE(I)=1./BIG
146 --- 115     CONTINUE
147 ---          LOW=M
148 ---          NM1=N-1
149 ---          DO 125 K=1,NM1
150 ---          LOW=MINO(LOW+1,N)
151 ---          BIG=0.
152 ---          DO 117 I=K,LOW
153 ---          SIZE=ABS(S(I,1))*SCALE(I)
154 ---          IF(SIZE-BIG) 117,117,116
155 --- 116     BIG=SIZE
156 ---          IPIV=I
157 --- 117     CONTINUE
158 ---          IF(BIG) 119,118,119
159 --- 118     WRITE(6,917)
160 --- 917     FORMAT('  IN BDSLV &&&&&&&&   ZERO PIVOT ELEMENT')
161 ---          NFLAG=1
162 ---          RETURN
163 --- 119     INDEX(K)=IPIV
164 ---          IF(IPIV-K) 120,122,120
```

```
    165 --- 120    SCALE(IPIV)=SCALE(K)
    166 ---        DO 121 J=1,IBW
    167 ---        TEMP=S(K,J)
    168 ---        S(K,J)=S(IPIV,J)
    169 ---        S(IPIV,J)=TEMP
    170 --- 121    CONTINUE
    171 --- 122    PIVOT=S(K,1)
    172 ---        KOUNT=0
    173 ---        KP1=K+1
    174 ---        DO 124 I=KP1,LOW
    175 ---        KOUNT=KOUNT+1
    176 ---        FACT=-S(I,1)/PIVOT
    177 ---        MUL(KOUNT,K)=FACT
    178 ---        DO 123 J=2,IBW
    179 ---        S(I,J-1)=S(I,J)+FACT*S(K,J)
    180 --- 123    CONTINUE
    181 ---        S(I,IBW)=0.
    182 --- 124    CONTINUE
    183 --- 125    CONTINUE
    184 ---        IF(S(N,1)) 126,118,126
    185 --- 126    DO 127 I=1,N
    186 ---        X(I)=B(I)
    187 --- 127    CONTINUE
    188 ---        LOW=M
    189 ---        DO 130 K=1,NM1
    190 ---        IPIV=INDEX(K)
    191 ---        IF(IPIV.EQ.K) GO TO 128
    192 ---        TEMP=X(K)
    193 ---        X(K)=X(IPIV)
    194 ---        X(IPIV)=TEMP
    195 --- 128    KOUNT=0
    196 ---        KP1=K+1
    197 ---        LOW=MINO(LOW+1,N)
    198 ---        DO 129 I=KP1,LOW
    199 ---        KOUNT=KOUNT+1
    200 ---        X(I)=X(I)+MUL(KOUNT,K)*X(K)
    201 --- 129    CONTINUE
    202 --- 130    CONTINUE
    203 ---        X(N)=X(N)/S(N,1)
    204 ---        LOW=1
    205 ---        DO 132 IBACK=I,NM1
    206 ---        I=N-IBACK
    207 ---        SUM=0.
    208 ---        LOW=MINO(LOW+1,IBW)
    209 ---        DO 131 JBACK=2,LOW
    210 ---        J=I+JBACK-1
    211 ---        SUM=SUM+S(I,JBACK)*X(J)
    212 --- 131    CONTINUE
    213 ---        X(I)=(X(I)-SUM)/S(I,1)
    214 --- 132    CONTINUE
    215 ---        RETURN
    216 ---        END

COMPILATION DATASET IS ABOUT TO BE WRITTEN
```

```
*EDIT,D=PLIB,DN=GTS3S,DL
      1 ---        SUBROUTINE GTS3S(IYD,SEC,ALT,GLAT,GLONG,STL,F107A,F107,AP,MASS,D, C
      2 ---       >T)
      3 --- C         A.E.HEDIN 1/2/76; UPDATED 3/8/76; 3/19/76 (F); 7/21/76 (AP)    C
      4 --- C         MSIS NEUTRAL THERMOSPHERE MODEL                                C
      5 --- C         ND LONGITUDINAL VARIATIONS                                     C
      6 --- C       INPUT:                                                           C
      7 --- C          IYD - YEAR AND DAY AS YYDDD                                   C
      8 --- C          SEC - UT(SEC)                                                 C
      9 --- C          ALT - ALTITUDE(KM) (GREATER THAN 120 KM)                      C
     10 --- C          GLAT - GEODETIC LATITUDE(DEG)                                 C
     11 --- C          GLONG - GEODETIC LONGITUDE(DEG)                               C
     12 --- C          STL - LOCAL APPARENT SOLAR TIME(HRS)                          C
     13 --- C          F107A - 3 MONTH AVERAGE OF F10.7 FLUX                         C
     14 --- C          F107 - DAILY F10.7 FLUX FOR PREVIOUS DAY                      C
     15 --- C          AP - MAGNETIC INDEX(DAILY)                                    C
     16 --- C          MASS - MASS NUMBER (ONLY DENSITY FOR SELECTED GAS IS          C
     17 --- C                CALCULATED UNLESS MASS = 48.  MASS 0 IS TEMPERATURE.)  C
     18 --- C       OUTPUT:                                                          C
     19 --- C          D(1)/SH(1) - HE NUMBER DENSITY(CM-3)/SCALE HEIGHT(KM)
     20 --- C          D(2) - O NUMBER DENSITY(CM-3)                                 C
     21 --- C          D(3)/SH(3) - N2 NUMBER DENSITY(CM-3)/SCALE HEIGHT(KM)
     22 --- C          D(4) - O2 NUMBER DENSITY(CM-3)                                C
     23 --- C          D(5)/SH(5) - AR NUMBER DENSITY(CM-3)/SCALE HEIGHT(KM)
     24 --- C          D(6) - TOTAL MASS DENSITY(GM/CM3)                             C
     25 --- C          D(7)/SH(7) - H NUMBER DENSITY(CM-3)/SCALE HEIGHT(KM)
     26 --- C          T(1) - EXOSPHERIC TEMPERATURE                                 C
     27 --- C          T(2) - TEMPERATURE AT ALT                                     C
     28 ---        DIMENSION SH(7)
     29 ---        COMMON/GTS3C/TLB,S,DB4,DB16,DB28,DB32,DB40,DB48,DB1              C
     30 ---        DIMENSION D(7),T(2),MT(8),DL(9)                                  C
     31 ---        DIMENSION   PT1(50),PT2(50),PA1(50),PA2(50),PB1(50),PB2(50),     C
     32 ---       * PC1(50),PC2(50),PD1(50),PD2(50),PE1(50),PE2(50)                 C
     33 --- C          TEMPERATURE                                                   C
     34 ---        DATA PT1/                                                        C
     35 ---       *  1.04130E 03, 2.59903E-03, 3.33263E-02,-1.23860E-01, 4.38076E-03,C
     36 ---       * -3.74330E-03,-1.06394E-01, 2.09679E-03, 5.43158E-03,-1.29812E-01,C
     37 ---       * -2.52187E-02, 1.17557E-02,-6.39594E-03,-1.14298E 01, 0.0        ,C
     38 ---       *  6.27580E-03, 7.29032E-03, 9.90241E 01, 1.12475E-02, 1.19836E-03,C
     39 ---       * -5.37448E-06, 3.19749E-03, 0.0        , 3.97740E-03, 1.00000E 00,C
     40 ---       *  0.0        , 0.0        , 0.0        , 0.0        , 1.00000E 00,C
     41 ---       *  0.0        ,-2.69558E 01, 5.92924E-03, 1.33886E-03, 1.00000E 00,C
     42 ---       *  1.12319E-02,-6.74976E-03,-1.61105E-02, 3.21281E 01, 8.58530E-04,C
     43 ---       *  2.87322E-04, 0.0        , 0.0        , 5.57165E-01, 1.83722E-01,C
     44 ---       *  9.05630E-03, 0.0        , 5.62044E-03, 0.0        , 5.58163E-03/C
     45 ---        DATA PT2/                                                        C
     46 ---       *  1.73064E-01,-3.73106E-02, 3.57049E-02,-1.49970E-02, 2.21841E-02,C
     47 ---       *  4.77211E-03, 3.74408E-02,-1.00145E-02, 8.16843E-03, 5.51251E-03,C
     48 ---       *  0.0        , 0.0        , 1.05642E 00, 0.0        , 0.0        ,C
     49 ---       *  0.0        , 1.66941E-03,-5.21344E-04, 1.71912E-03, 4.54863E-04,C
     50 ---       *  1.82705E-03, 5.60343E-03, 2.94944E-03,-2.37463E-03, 9.74066E-02,C
     51 ---       *  2.49258E-01, 0.0        , 0.0        , 0.0        , 0.0        ,C
     52 ---       *  0.0        , 0.0        , 0.0        , 0.0        , 0.0        ,C
     53 ---       *  0.0        , 0.0        , 0.0        , 0.0        , 0.0        ,C
     54 ---       *  0.0        , 0.0        , 0.0        , 0.0        , 0.0        ,C
```

```
     55 ---        *  0.0          , 0.0          , 0.0          , C.C          , C.C          /(
     56 --- C           HE DENSITY                                                             (
     57 ---         DATA PA1/                                                                 (
     58 ---        *  1.00000E 00,-1.13108E-01,-1.99552E-01,-2.62923E-01,-2.72419E-02,(
     59 ---        *  3.70448E-02, 4.98992E-01, 6.11127E-03,-3.82857E-02, 1.66725E 00,(
     60 ---        *  2.53110E-01, 1.42720E-01, 1.31387E-01,-7.33643E 00, C.C          ,(
     61 ---        *  3.52942E-01,-8.84136E-02, 1.06182E 02,-1.14466E-01,-6.92663E-04,(
     62 ---        *  0.0          , 0.0          , 0.0          , C.C          , 1.00000E 00,(
     63 ---        *  0.0          ,-1.09623E-01,-3.19738E-02,-3.44951E-02, C.C          ,(
     64 ---        *  C.0          ,-1.80985E 02,-9.56827E-03, C.C          , 1.00000E 00,(
     65 ---        *  6.28148E-02,-2.77382E-02, 9.66159E-02,-3.99226E 01, C.C          ,(
     66 ---        *  0.C          , 0.C          , 0.0          , 3.96583E-01, 1.93782E-01,(
     67 ---        * -6.41994E-02, 0.C          , 1.37334E-04, C.C          , 6.40007E-03/(
     68 ---         DATA PA2/
     69 ---        * -3.97809E-01,-9.14425E-01, 5.10801E-02, C.C          , C.C          ,(
     70 ---        *  1.00000E 00, 0.0          , 0.0          , C.C          , C.C          ,(
     71 ---        * -3.28663E-01, 3.26960E-01, 1.25265E 00, 1.42507E-02,-1.42531E-02,(
     72 ---        * -1.55744E-03, 4.66277E-03, 1.12219E-02,-3.15129E-02,-3.02730E-02,(
     73 ---        * -5.75410E-03,-1.87383E-02,-1.72977E-02, 3.56028E-03, 6.61580E-02,(
     74 ---        *  7.44005E-02, 0.0          , 0.0          , C.0          , C.C          ,(
     75 ---        *  0.C          , 0.C          , 0.0          , C.C          , C.C          ,(
     76 ---        *  0.0          , 0.0          , 0.0          , C.C          , C.C          ,(
     77 ---        *  0.0          , 0.0          , 0.0          , C.C          , C.C          ,(
     78 ---        *  C.0          , 0.0          , 0.0          , C.C          , C.C          /(
     79 --- C           B DENSITY                                                              (
     80 ---         DATA PB1/                                                                 (
     81 ---        *  1.00000E 00,-1.04704E-01,-1.25208E-01,-9.34052E-02,-2.99355E-02,(
     82 ---        * -1.02674E-02, 1.51211E-01,-2.73808E-02,-1.69264E-02, 4.21340E-01,(
     83 ---        *  C.0          , 2.21850E-02, 2.91846E-02,-1.60451E 01, C.C          ,(
     84 ---        *  1.37033E-01, 0.0          , 1.04246E 02, 6.47395E-02, 9.58775E-04,(
     85 ---        *  C.0          , 2.26063E-03, 0.0          , C.C          , 1.00000E 00,(
     86 ---        *  0.0          ,-3.84385E-02, 0.0          , C.C          , 1.00000E 00,(
     87 ---        *  0.0          ,-4.33849E 01,-9.34153E-04, C.C          , 1.00000E 00,(
     88 ---        *  0.0          , 0.0          ,-7.53599E-02,-3.58478E 01, C.C          ,(
     89 ---        *  0.0          , 0.0          , 0.0          , 6.64041E-01, 2.75610E-01,(
     90 ---        * -3.27549E-02, C.0          , 1.89232E-03, C.C          , 2.16012E-03/(
     91 ---         DATA PB2/                                                                 (
     92 ---        * -1.13167E 00,-1.18027E 00, 2.00554E-02, C.C          , C.C          ,(
     93 ---        *  0.0          , 0.0          , 0.0          ,-9.64905E-01,-4.57907E-01,(
     94 ---        * -5.49877E-01,-2.12852E-02, 9.47113E-01, 8.14077E-01,-7.77800E-03,(
     95 ---        *  2.03634E-02, 3.02358E-03, 1.90946E-03, 2.62931E-03, 2.87621E-03,(
     96 ---        * -7.27535E-03,-1.88626E-02,-1.30574E-02, 6.30837E-03,-4.56661E-02,(
     97 ---        *  4.20439E-01, 0.0          , 0.0          , C.0          , C.C          ,(
     98 ---        *  0.C          , 0.0          , 0.0          , C.C          , C.C          ,(
     99 ---        *  0.0          , 0.0          , 0.0          , C.0          , C.C          ,(
    100 ---        *  0.0          , 0.0          , 0.0          , C.0          , C.0          ,(
    101 ---        *  0.0          , 0.0          , 0.0          , C.C          , C.C          /(
    102 --- C           O2 DENSITY                                                             (
    103 ---         DATA PC1/                                                                 (
    104 ---        *  1.00000E 00, 0.0          , 0.0          ,-1.42845E-01, C.C          ,(
    105 ---        *  2.89324E-02, 2.38569E-01, 0.0          ,-3.05069E-04, 2.31082E-01,(
    106 ---        *  C.0          , 0.C          , 0.0          ,-9.91794E-01, C.C          ,(
    107 ---        * -2.17693E-01, 0.0          , 9.56118E 01,-1.70944E-01, C.C          ,(
    108 ---        *  C.0          , C.0          , 0.0          ,-1.73501E-02, C.C          ,(
    109 ---        *  0.0          , 0.0          , 0.0          , C.C          , C.C          ,(
```

```
110 ---       *  C.0          ,-2.20981E 01,-5.92085E-03,-3.37127E-03, 9.35465E-01,C
111 ---       *  0.0          , 0.0         ,-1.75075E-01, 2.08943E 01, C.C         ,C
112 ---       *  0.0          , 0.0         , 0.0        , 2.50000E-01, 1.00000E 00,C
113 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        /C
114 ---        DATA PC2/                                                            C
115 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
116 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
117 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
118 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
119 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
120 ---       *  C.0          , 0.0         , 0.0        , C.C        , C.C        ,C
121 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
122 ---       *  C.0          , 0.0         , 0.0        , C.C        , C.C        ,C
123 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
124 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        /C
125 --- C           AR DENSITY                                                     C
126 ---        DATA PD1/                                                           C
127 ---       *  1.00000E 00, 0.0           , 0.0        ,-6.43002E-02, C.C        ,C
128 ---       * -1.72016E-02,-5.08900E-02, C.0          ,-6.48594E-02,-2.24349E-01,C
129 ---       *  0.0          , 1.10820E-01,-9.22838E-02, 8.40296E 00, C.C        ,C
130 ---       * -3.21876E-02, 1.66617E-01, 5.63081E 01,-6.41921E-02,-3.88443E-03,C
131 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
132 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
133 ---       *  C.0          , 6.17283E 01, 4.73491E-03, C.0        , 1.00000E 00,C
134 ---       *  0.0          , 0.0         ,-1.74916E-01,-1.09277E 01, C.C        ,C
135 ---       *  0.0          , 0.0         , 0.0        , 2.50000E-01, 1.00000E 00,C
136 ---       *  1.16579E-02, 0.0           , 0.0        , C.C        , C.C        /C
137 ---        DATA PD2/                                                           C
138 ---       *  C.0          , 0.0         , 0.0        , C.C        , C.C        ,C
139 ---       *  1.00000E 00, C.C           , 0.0        , C.C        , C.C        ,C
140 ---       *  1.87598E 00, 1.08431E 00, 1.13167E 00, C.0        , C.C        ,C
141 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
142 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
143 ---       *  C.0          , 0.0         , 0.0        , C.C        , C.C        ,C
144 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
145 ---       *  0.0          , C.0         , 0.0        , C.0        , C.C        ,C
146 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
147 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        /C
148 --- C             H DENSITY
149 ---        DATA PE1/
150 ---       *  1.00000E 00,-1.33654E-01, 0.0          , 2.09446E-01, 2.82952E-02,
151 ---       *  0.0          , 3.83910E-01, 2.94125E-02, C.C        , 3.31088E-01,
152 ---       *  1.04532E-01,-2.47471E-02, 4.35616E-02,-1.47746E 01, C.C        ,
153 ---       *  0.0          , 0.0         , 9.82546E 01, 5.87827E-02,-1.24608E-02,
154 ---       *  C.0          ,-1.93000E-02, 0.0          , C.C        , C.C        ,
155 ---       *  0.0          , 0.0         , 0.0        , C.0        , C.C        ,
156 ---       *  C.0          , 9.22737E 01,-5.18541E-03, C.0        , 1.C        ,
157 ---       *  8.57138E-02,-3.97449E-03,-9.06507E-02,-7.20036E 01, C.C        ,
158 ---       *  C.0          , 0.0         , 0.0        , 2.50000E-01, 1.00000E 00,
159 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        /C
160 ---        DATA PE2/                                                           C
161 ---       *  C.0          , 0.0         , 0.0        , C.C        , C.C        ,C
162 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
163 ---       *  C.0          , 0.0         , 0.0        , C.0        , C.C        ,C
164 ---       *  0.0          , 0.0         , 0.0        , C.C        , C.C        ,C
```

```
165 ---       *  0.0          , 0.0          , 0.0          , C.0          , C.C          ,C
166 ---       *  C.C          , 0.0          , 0.0          , C.C          , C.C          ,C
167 ---       *  C.0          , C.0          , 0.0          , C.0          , C.C          ,C
168 ---       *  0.0          , 0.0          , 0.0          , C.C          , C.C          ,C
169 ---       *  C.C          , 0.0          , 0.0          , C.0          , C.C          ,C
170 ---       *  0.0          , 0.0          , 0.0          , C.C          , C.C          /C
171 --- C           LOWER BOUNDARY                                                        C
172 ---        DATA DL/                                                                   C
173 ---       *  1.20000E 02, 3.86000E 02, 2.54800E-02, 2.45600E 07, 8.59400E 10,C
174 ---       *  2.81000E 11, 3.30000E 10, 1.33000E 09, 1.76100E 05/                     C
175 ---        DATA MT/48,0,4,16,28,32,40,1/                                             C
176 ---        YRD=IYD
177 ---        T(1) = GLOBE(YRD,SEC,GLAT,GLONG,STL,F107A,F107,AP,PT1(1),PT2(1))          C
178 ---        TLB=GLB(SR,G28)*DL(2)                                                     C
179 ---        S = SR*DL(3)                                                              C
180 ---        DO 10 J = 1,8                                                             C
181 ---        IF(MASS.EQ.MT(J))   GO TO 15                                              C
182 ---     10 CONTINUE                                                                  C
183 ---        WRITE(6,100) MASS                                                         C
184 ---        GO TO 90                                                                  C
185 ---     15 GO TO (20,50,20,25,30,35,40,45),  J                                       C
186 ---     20 GC=DCR(ALT,PA1(1),T(1))                                                   C
187 ---        G4 = GLOB(PA1(1),PA2(1))                                                  C
188 ---        DB4 = DL(4) *EXP(G4-1.)                                                   C
189 ---        D(1) = DENSS(ALT,DB4,T(1),TLB, 4.,-.4,T(2),DL(1),S,SH(1))
190 ---        IF(MASS.NE.48)   GO TO 90                                                 C
191 ---     25 GG=DCR(ALT,PB1(1),T(1))
192 ---        G16 = GLOB(PB1(1),PB2(1))                                                 C
193 ---        DB16 =  DL(5)*EXP(G16-1.)                                                 C
194 ---        D(2) = DENSS(ALT,DB16,T(1),TLB, 16.,0.,T(2),DL(1),S,SH(2))
195 ---        IF(MASS.NE.48)   GO TO 90                                                 C
196 ---     30 DB28 = DL(6)*EXP(G28-1.)                                                  C
197 ---        D(3) = DENSS(ALT,DB28,T(1),TLB, 28.,0.,T(2),DL(1),S,SH(3))
198 ---        IF(MASS.NE.48)   GO TO 90                                                 C
199 ---     35 G32 = GLOB(PC1(1),PC2(1))                                                 C
200 ---        DB32 = DL(7)*EXP(G32-1.)                                                  C
201 ---        D(4) = DENSS(ALT,DB32,T(1),TLB, 32.,0.,T(2),DL(1),S,SH(4))
202 ---        IF(MASS.NE.48)   GO TO 90                                                 C
203 ---     40 G40=GLOB(PD1(1),PD2(1))                                                   C
204 ---        DB40 = DL(8)*EXP(G40-1.)                                                  C
205 ---        D(5) = DENSS(ALT,DB40,T(1),TLB, 40.,0.,T(2),DL(1),S,SH(5))
206 ---        IF(MASS.NE.48)   GO TO 90                                                 C
207 ---     45 G1 = GLOB(PE1(1),PE2(1))                                                  C
208 ---        DB1 = DL(9)*EXP(G1-1.)                                                    C
209 ---        D(7)=DENSS(ALT,DB1,T(1),TLB,1.,-.4,T(2),DL(1),S,SH(7))
210 ---        IF(MASS.NE.48) GO TO 90                                                   C
211 ---        D(6) = 1.66E-24*(4.*D(1)+16.*D(2)+28.*D(3)+32.*D(4)+40.*D(5)+D(7))C
212 ---        DB48=1.66E-24*(4*DB4+16.*DB16+28.*DB28+32.*DB32+40.*DB40+DB1)
213 ---        GO TO 90                                                                  C
214 ---     50 D(1) = DENSS(ALT,1., T(1),TLB,0.,C.,T(2),DL(1),S,SH(1))
215 ---     90 RETURN                                                                    C
216 ---    100 FORMAT(1X,*MASS*, I5, *  NOT VALID*)                                      C
217 ---        END                                                                       C
218 ---        FUNCTION GLOBE(YRD,SEC,LAT,LONG,TLOC,F107A,F107,AP,P,C)                   C
219 --- C   FILE GLOBA                                                                  C
```

```
220 ---         ENTRY GLOB(P,Q)                                                  C
221 ---         REAL LAT, LONG                                                   C
222 ---         DIMENSION P(1),PLG(9,4),PLM(8,4),Q(1),SW(15),              AP(1) C
223 ---         DIMENSION DC(10)                                                 C
224 ---         DIMENSION SV(15)
225 ---         DIMENSION PL(50),KPL(50),KPP(50)
226 ---         COMMON/TTEST/T(15)                                               C
227 ---         DATA DGTR/1.74533E-2/,DP/1.72142E-2/, XL/1000./,TLL/1000./       C
228 ---         DATA IDC/0/,DAYL/-1./                                            C
229 ---         DATA HR/.2618/, SR/7.2722E-5/,SW/15*1.     /,NSW/12/,SECL/100./  C
230 ---         T(10) = 0.                                                       C
231 ---         T(11) = 0.                                                       C
232 ---         T(12) = 0.                                                       C
233 ---         IF(IDC.GE.1) GO TO 10                                            C
234 ---         DO 5 I=1,10                                                      C
235 ---       5 DC(I)=1.                                                         C
236 ---         IDC=2                                                            C
237 ---      10 CONTINUE                                                         C
238 ---         IYR = YRD/1000.                                                  C
239 ---         DAY = YRD - IYR*1000.                                            C
240 ---         IF(SW(10).EQ.-1.)   GO TO 40                                     C
241 ---         IF(XL.EQ.LAT)   GO TO 15                                         C
242 ---         C = SIN(LAT*DGTR)                                                C
243 ---         S = COS(LAT*DGTR)                                                C
244 ---         C2 = C*C                                                         C
245 ---         C4 = C2*C2                                                       C
246 ---         S2 = S*S                                                         C
247 ---         PLG(2,1) = C                                                     C
248 ---         PLG(3,1) = 0.5*(3.*C2 -1.)                                       C
249 ---         PLG(4,1) = 0.5*(5.*C*C2-3.*C)                                    C
250 ---         PLG(5,1) = (35.*C4 - 30.*C2 + 3.)/8.                             C
251 ---         PLG(6,1) = (63.*C2*C2*C - 70.*C2*C + 15.*C)/8.                   C
252 --- C       PLG(7,1) = (11.*C*PLG(6,1) - 5.*PLG(5,1))/6.                     C
253 --- C       PLG(8,1) = (13.*C*PLG(7,1) - 6.*PLG(6,1))/7.                     C
254 ---         PLG(2,2) = S                                                     C
255 ---         PLG(3,2) = 3.*C*S                                                C
256 ---         PLG(4,2) = 1.5*(5.*C2-1.)*S                                      C
257 ---         PLG(5,2) = 2.5*(7.*C2*C-3.*C)*S                                  C
258 ---         PLG(6,2) = 1.875*(21.*C4 - 14.*C2 +1.)*S                         C
259 --- C       PLG(7,2) = (11.*C*PLG(6,2)-6.*PLG(5,2))/5.                       C
260 --- C       PLG(8,2) = (13.*PLG(7,2)-7.*PLG(6,2))/6.                         C
261 --- C       PLG(9,2) = (15.*C*PLG(8,2)-8.*PLG(7,2))/7.                       C
262 ---         PLG(3,3) = 3.*S2                                                 C
263 ---         PLG(4,3) = 15.*S2*C                                              C
264 ---         PLG(5,3) = 7.5*(7.*C2 -1.)*S2                                    C
265 --- C       PLG(6,3) = 3.*C*PLG(5,3)-2.*PLG(4,3)                             C
266 ---         PLG(4,4) = 15.*S2*S                                              C
267 --- C       PLG(5,4) = 105.*S2*S*C                                           C
268 ---      15 CONTINUE                                                         C
269 ---         IF(TLL.EQ.TLOC)   GO TO 16                                       C
270 ---         STLOC = SIN(HR*TLOC)                                             C
271 ---         CTLOC = COS(HR*TLOC)                                             C
272 ---         S2TLOC = SIN(2.*HR*TLOC)                                         C
273 ---         C2TLOC = COS(2.*HR*TLOC)                                         C
274 ---         S3TLOC = SIN(3.*HR*TLOC)                                         C
```

```
      C3TLOC = COS(3.*HR*TLOC)                                          C
      TLL = TLOC                                                        C
   16 CONTINUE                                                          C
      IF(DAY.NE.DAYL.OR.P(14).NE.P14) CD14=COS(DR*(DAY-P(14)))          C
      IF(DAY.NE.DAYL.OR.P(18).NE.P18) CD18=COS(2.*DR*(DAY-P(18)))       C
      IF(DAY.NE.DAYL.OR.P(32).NE.P32) CD32=COS(DR*(DAY-P(32)))          C
      IF(DAY.NE.DAYL.OR.P(39).NE.P39) CD39=COS(2.*DR*(DAY-P(39)))       C
      DAYL = DAY                                                        C
      P14 = P(14)                                                       C
      P18 = P(18)                                                       C
      P32 = P(32)                                                       C
      P39 = P(39)                                                       C
C          F10.7 EFFECT                                                 C
      DF = F107 - F107A                                                 C
      DFA=F107A-150.                                                    C
      T(1) =  (P(20)*DF + P(21)*DF*DF + P(22)*(F107A-150.))*SW(1)       C
      F1 = 1. + SW(1)*(P(48)*DFA+P(47)*DF)                              C
      F2=1.+ SW(1)*(P(50)*DFA+P(49)*DF)                                 C
C        TIME INDEPENDENT                                               C
      T(2) =                                                            C
     1  (P(2)*PLG(3,1) + P(3)*PLG(5,1))                                 C
     2 +P(27)*PLG(2,1)                                                  C
C          SYMMETRICAL ANNUAL                                           C
      T(3) =                                                            C
     1 (P(19)+P(23)*PLG(3,1))*CD32                                      C
C          SYMMETRICAL SEMIANNUAL                                       C
      T(4) =                                                            C
     1 (P(16)+P(17)*PLG(3,1))*CD18*DC(6)                                C
C          ASYMMETRICAL ANNUAL                                          C
      T(5) =  F1*                                                       C
     1  (P(10)*PLG(2,1) + P(11)*PLG(4,1) +P(26)*PLG(6,1))*CD14*DC(5)    C
C           ASYMMETRICAL SEMIANNUAL                                     C
      T(6) =    P(38)*PLG(2,1)*CD39                                     C
C          DIURNAL                                                      C
      T71 = P(12)*PLG(3,2)*CD14 + P(36)*PLG(2,2)*CD14                   C
      T72 = P(13)*PLG(3,2)*CD14 + P(37)*PLG(2,2)*CD14                   C
      T(7) = F2*                                                        C
     1 ((P(4)*PLG(2,2) + P(5)*PLG(4,2) + P(28)*PLG(6,2)                 C
     2 + T71)*CTLOC*DC(1)                                               C
     4 + (P(7)*PLG(2,2) + P(8)*PLG(4,2) +P(29)*PLG(6,2)                 C
     5 + T72)*STLOC*DC(2))                                              C
C          SEMIDIURNAL + TERDIURNAL                                     C
      T81 = P(24)*PLG(4,3)*CD14                                         C
      T82 = P(34)*PLG(4,3)*CD14                                         C
      T(8) = F2*                                                        C
     1 ((P(6)*PLG(3,3) + T81)*C2TLOC*DC(3)                              C
     3 +(P(9)*PLG(3,3) + T82)*S2TLOC*DC(4)                              C
     4 +     (P(40)*PLG(4,4)                        )*S3TLOC            C
     5 +(P(41)*PLG(4,4)                          )*C3TLOC)              C
C           MAGNETIC ACTIVITY BASED ON DAILY AP                         C
      APD=(AP(1)-4.)                                                    C
      APDF = (APD+(P(45)-1.)*(APD+(EXP(-P(44)*APD)-1.)/P(44)))          C
      T(9) =APDF*(P(33)+P(46)*PLG(3,1))*DC(7)                           C
      IF(IDC.EQ.1) IDC=0                                                C
C FILE GLOB3                                                            C
```

```
      40 TINF = 1.
         DO 50 I = 1,NSW
      50 TINF = TINF + ABS(SW(I))*T(I)
         GLOBE = TINF*P(1)
         XL = LAT
         SECL =SEC
         XLON = LONG
         RETURN
         ENTRY TSELEC(SV)
         DO 100 I = 1,15
     100 SW(I) = SV(I)
         RETURN
         ENTRY GLB(SS,DR28)
         DATA PL/4*0.,1.14E-3,-1.27E-3,-2.55E-2,7.38E-3,-.037,-1.2CE-1,
        $  .013,6.70E-2,2.73E-3,10*0.,1.39E-3,25*0./
         DATA NPL/10/,KPL/14,15,16,17,18,19,2C,21,22,23,24,25,26/
         DATA KPP/51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66/
         GLOBE =   P(30)
         SS=     P(25)
         DR28 = 1.
         IF(SW(14).EQ.0.) RETURN
         IF(NPL.EQ.0)  GO TO 201
         DO 200 L=1,NPL
     200 PL(KPL(L))=P(KPP(L))
     201 CONTINUE
C            LOWER BOUND TEMPERATURE
         GLOBE=1.+
        $ SW(5)*(PL(9)*PLG(2,1)                    )*CD14+
        $ SW(4)*(PL(11)                  )*CD18
         GLOBE=  GLOBE*P(30)
C            SHAPE FACTOR
         SS=    P(25)*(1.
        1+SW(7)*((PL(14)*PLG(2,2)+PL(15)*PLG(4,2)+PL(22)*PLG(6,2))*CTLOC
        2       +(PL(16)*PLG(2,2)+PL(17)*PLG(4,2)+PL(23)*PLG(6,2))*STLOC)
        3 +SW(8)*((PL(18)*PLG(3,3)+PL(19)*PLG(5,3))*C2TLOC
        4        +(PL(2C)*PLG(3,3)+PL(21)*PLG(5,3))*S2TLOC))
C
C       LOWER BOUND N2 RATIO
         DR28=1.
        1+SW(7)*((PL(1)*PLG(2,2)+PL(2)*PLG(4,2))*CTLOC
        2       +(PL(3)*PLG(2,2)+PL(4)*PLG(4,2))*STLOC)
        3 +SW(8)*((PL(5)*PLG(3,3)+PL(6)*PLG(5,3))*C2TLOC
        4        +(PL(7)*PLG(3,3)+PL(8)*PLG(5,3))*S2TLOC)
        $ + SW(9)*APD*(PL(13)+PL(24)*PLG(3,1))
        $ +SW(5)*PL(10)*PLG(2,1)*CD14
        $ +SW(4)*PL(12)*CD18
         RETURN
         ENTRY DCR(AL ,P, TNF)
         IF(SW(15).EQ.0. .OR. P(53).EQ.0.) RETURN
         ALT=AL
         IF(AL.LT.140.) ALT=140.
         EXPD=EXP((15C.-ALT)*ABS(P(53)))
         DC(1)=1.+P(51)*EXPD
         DC(2)=1.+P(52)*EXPD
         DC(3)=1.+P(54)*EXPD
```

```
    385 ---         DC(4)=1.+P(54)*EXPD                                                  (
    386 ---         DC(5)=1.+P(57)*EXPD                                                  (
    387 ---         DC(6)=1.+P(58)*EXPD                                                  (
    388 ---         DC(7)=1.+P(59)*EXPD                                                  (
    389 ---         IDC=1                                                                (
    390 ---         RETURN                                                               (
    391 ---         END                                                                  (
    392 --- C.....................................................................
    393 ---         FUNCTION DENSS(ALT,DLB,TINF,TLB,MW,ALPHA,TZ,ZLB,S,SH)
    394 --- C           CALCULATE STATIC DIFFUSION MODEL AMBIENT DENSSITIES
    395 --- C               ALT = ALTITUDE
    396 --- C               DLB = DENSSITY AT LOWER BOUNDARY
    397 --- C               TINF = EXOSPHERE TEMPERATURE
    398 --- C               TLB = TEMPERATURE AT LOWER BOUNDARY
    399 --- C               MW = MOLECULAR WEIGHT
    400 --- C               ALPHA = THERMAL DIFFUSION COEFF.
    401 --- C              ZLB = ALTITUDE OF LOWER BOUNDARY
    402 --- C              TZ = TEMP AT ALT (OUTPUT)
    403 --- C               S = TEMPERATURE GRADIENT PARAMETER
    404 --- C                 SH=SCALE HT AT ALT(KM)
    405 ---         REAL MW
    406 ---         DATA RE/6356.77/,GSURF/980.665/,RGAS/8.314E+7/
    407 ---         DENSS = 1.
    408 ---         SIGMA = S + 1./(RE+ZLB )
    409 ---         ZET = (RE+ZLB)/(RE+ALT)
    410 ---         ZETA=(ALT-ZLB)*ZET
    411 ---         EXPSZ = EXP(-SIGMA*ZETA)
    412 ---         TZ = TINF - (TINF-TLB)*EXPSZ
    413 ---         IF(MW .EQ. 0.) GO TO 10
    414 ---         GLBS    = GSURF/(1.+ZLB/RE)**2
    415 ---         A = 1. - TLB/TINF
    416 ---         GAMMA = MW*GLBS/(SIGMA*RGAS*TINF)
    417 ---         SI=ZET**2*SIGMA*(GAMMA+(1.0+GAMMA+ALPHA)*A*EXPSZ/(1.-A*EXPSZ))
    418 ---         SH=1./SI
    419 ---         DENSS = DLB*((1.-A)/(1.-A*EXPSZ))**(1.+ALPHA+GAMMA)
    420 ---        1 *EXP(-SIGMA*GAMMA*ZETA)
    421 ---       10 CONTINUE
    422 ---         RETURN
    423 ---         END

TINATION DATASET IS ABOUT TO BE WRITTEN

RF GIVEN,  RF DONE
TINATION DATASET HAS NOW BEEN WRITTEN

+-+-+-+-+-+-+-+-+-+    EDITOR TERMINATING                    41/100 SECONDS IS ELAPSED
```

```
- $EDIT,D=PLIB,DN=RSPR,DL
      1 --- C..............<RSPR>.................
      2 ---            SUBROUTINE PSPR(Z,ZDX,ZN2,ZO2,SZA,TN,ZPR,JMAX,SW1,TPRION,EM)
      3 --- C$$$$  PROGRAM FOR THE CALCULATION OF THE PHOTOELECTRON ENERGY
      4 --- C$$$$  SPECTRUM, AT SPECIFIED ALTITUDES. THIS VERSION USES ARRAY ELSPEC
      5 --- C$$$$  TO CREATE A DATA FILE FOR THE 2-STREAM PROGRAM
      6 --- C.... THIS PROGRAM COMBINES A A. NAGY PROGRAM WITH THE C. TORR CELLMA DEP
      7 --- C.... ZPR IS THE UPPER ALTITUDE FOR PRODUCTION, BUT THE
      8 --- C.... PROGRAM WILL NOT ALLOW > 60 POINTS ON EACH END OF FIELD LINE
      9 --- C++++ SET DCB 9=PSPROD , ELEC SPEC ON DCB=8
     10 --- C++++ Z=ALTITUDE; ZDX,ZN2,ZO2,TN ARE NEUT DENS & TEMP ARRAYS
     11 --- C++++ SZA=SOLAR ZENITH ANGLE, ZPR=MAX ALT FOR PROD CALC, JMAX=
     12 --- C++++ TOTAL # OF ALT STEPS, SW1=SWITCH =0 CALC ION PROD ONLY
     13 --- C++++ TPRION=ION PRODUCTION
     14 ---            REAL N
     15 ---           DIMENSION ZFLUX(100),FLUX(100),SIGABS(3,100),SIGION(3,100),
     16 ---          1COLUMN(3),PROD(3),XN(3),SPECT(125),PROB(3,7,100),XNE(3),
     17 ---          1PHE1(73),PHE2(73),ENER(195),PR4(2),TPOT(3,7),NNN(3),TPRION(3,300)
     18 ---          1    ,EPSIL1(3,7,100),EPSIL2(3,7,100),DEL(125),PEHEAT(3,7)
     19 ---          3     ,ELSPEC(120,100),ZLAM(100),PRION(3,7)
     20 ---           DIMENSION ZDX(300),ZN2(300),ZO2(300),TN(300),SZA(300),Z(300)
     21 ---           INTEGER SW1,SW2
     22 --- C.... SWITCH ISW IS USED TO SUPPRESS SOME OUTPUT (=0/1)
     23 ---           DATA  ISW,  RE   ,  GE   ,LMAX,IJC,FAC
     24 ---          >    /    0 ,6.357E8, 980 , 0   , 0 , 1  /
     25 ---           WRITE(6,103)
     26 ---           IEQ=(JMAX+1)/2
     27 ---           IF(Z(JMAX).GE.ZPR) IEQ=JMAX+1
     28 ---           IJC=0
     29 ---           DO 33 I=1,3
     30 ---           DO 33 J=1,JMAX
     31 ---           IF(J.GE.IEQ) GO TO 33
     32 ---           IF(Z(J).LT.ZPR) JPR=J
     33 --- 33       TPRION(I,J)=0.0
     34 ---           IF(JPR.GT.60) JPR=60
     35 --- C##########      READ THE INPUT PARAMETERS     ##########
     36 ---           IF(LMAX.NE.0) GO TO 149
     37 --- 1000       READ(5,200) CHI,IW1,SW2,LMAX
     38 --- C.....  NNN = NO. OF IONIZATION STATES/SPECIES ; TPOT = IONIZ. POT
     39 ---          DATA NNN/5,7,4/
     40 ---          DO 91 K1=1,3
     41 ---          K2=NNN(K1)
     42 --- 91       READ(5,92) (TPOT(K1,L1),L1=1,K2)
     43 ---           DO 85 L=1,LMAX
     44 --- 85        READ(5,109)  ZLAM(L),PHE1(L),PHE2(L),ZFLUX(L),
     45 ---          1     (SIGABS(I,L),I=1,3),(SIGION(IL,L),IL=2,3)
     46 --- C****CHI=ZENITHANGLE(RAD), SW1IS SWITCH: FOR TOTAL IONIZATION ONLY **
     47 --- C****SET GREATER THAN ZERO; IMAX=ALT INCS FOR DENSITY DATA; LMAX ****
     48 --- C****=WAVELENGTH INCS; Z(1) IS LOWEST ALT; DELZ IS THE ALT ****
     49 --- C****INC JMAX= ALT INCS TO BE USED IN THE CALCS; ****
     50 --- C**** RE=EARTH RADIUS ; GE=GRAVITY AT SURFACE  ******
     51 --- C****!!!BE SURE THAT DENSITIES ARE PROVIDED DOWN TO GRAZING HEIGHT!!!
     52 --- C....  PHE1-2 ARE START & END OF WAVELENGTH BOX IN EV ; ZFLUX IS INTENSIT
     53 --- C...   ; SIGABS = ABSORB. COEFF/E-18 ;SIGION = TOTAL ION. X-SECTION
     54 ---          DO 789 LL=1,LMAX
```

```
      ZFLUX(LL)=ZFLUX(LL)*1.E9*FAC
      SIGION(I,LL)=SIGABS(I,LL)
      DO 789 II=1,3
      IF(SIGABS(II,LL).LT.SIGION(II,LL)) SIGABS(II,LL)=SIGION(II,LL)
      SIGION(II,LL)=SIGION(II,LL)*1.E-16
789   SIGABS(II,LL)=SIGABS(II,LL)*1.E-18
C..... EPSIL1-2 ARE  ENERGY P.E. *S RESULTING FROM A GIVEN STATE
      DO 93 L=1,LMAX
      DO 93 I=1,3
      K1=NNN(I)
      DO 93 K=1,K1
      EPSIL1(I,K,L)=PHE1(L)-TPOT(I,K)
      EPSIL2(I,K,L)=PHE2(L)-TPOT(I,K)
93    CONTINUE
C...  FIND PROBABILITY FOR FORMATION OF EACH STATE  ........
      CALL PROBS(PROB,ZLAM,LMAX)
       IF(ISW.EQ.0) GO TO 97
        DO 96 I=1,3
       NNNI=NNN(I)
96     WRITE(6,209) ((PROB(I,IN,L),L=1,LMAX),IN=1,NNNI)
97     CONTINUE
C++++  SETTING UP ENERGY BOXES (MIDPT. ENERGIES)  ++++++
      DATA DEL/125*1./
      ENER(1)=0.5
      DO 1492 M=2,125
1492  ENER(M)=ENER(M-1)+0.5*(DEL(M-1)+DEL(M))
C%%%%
C%%%%%% ALTITUDE LOOP BEGINS HERE  %%
C% PROD=ION PROD. ; PEHEAT=ELECTRON HEAT PROD. ;PRION=PROD. RATE
C%%  OF EACH ION STATE , IJC USED FOR CONJUGATE HEMISPHERE
149   DO 7 J=1,JMAX
       IJ=J+1JC
      DO 786 I=1,3
      PROD(I)=0.
      DO 786 K=1,7
      PRION(I,K)=0.0
786   PEHEAT(I,K)=0.
      TNJ=TN(IJ)
      XN(1)=ZOX(IJ)
      XN(2)=ZO2(IJ)
      XN(3)=ZN2(IJ)
      ZZ=Z(IJ)*1.0E+5
      CHI=SZA(IJ)
CEEEE EVALUATE THE NEUTRAL COLUMN DENSITY  EEEEEEEE
      CALL RCOLUM(CHI,ZZ,TNJ,XN,COLUMN,RE,GE)
   82 DO13 M= 1, 125
   13 SPECT(M)= 0.0
C------  WAVELENGTH LOOP BEGINS HERE  ----------
C     TAU = OPTICAL DEPTH ;FLUX = FLUX AT ALT. J & WAVELENGTH L
      DO 6 L=1,LMAX
      TAU=0.
      DO 4 I=1,3
4     TAU=TAU+SIGABS(I,L)*COLUMN(I)
      FLUX(L)=ZFLUX(L)*EXP(-TAU)
      PTOT=0.
```

```
110 ---  C***** SPECIES LOOP BEGINS HERE *****
111 ---        DO 304 I=1,3
112 ---        K1=NNA(I)
113 ---  C********** LOOP THRU* ENERGY STATES     **********
114 ---  C** DSPECT=# P.E.*S FORMED BY M-L L BY IONIZATION OF THE K STATE OF THE I
115 ---        DO 302 K=1,K1
116 ---        E1= EPSIL1(I,K,L)
117 ---        E2= EPSIL2(I,K,L)
118 ---        IF (E2 .LT. 0.) GO TO 302
119 ---        DSPECT=XN(I)*SIGION(I,L)*FLUX(L)*PROB(I,K,L)
120 ---         IF(E1.LT.0) DSPECT=DSPECT*E2/(E2-E1)
121 ---         IF(E1.LT.0)  E1=0
122 ---        Y= E2- E1
123 ---        PROD(I)=PROD(I)+DSPECT
124 ---        PRION(I,K)=PRION(I,K)+DSPECT
125 ---        PEHEAT(I,K)=PRION(I,K)*TPOT(I,K)
126 ---        PTOT=PTOT+PEHEAT(I,K)
127 ---  C;;;;;;     NOW DISTRIBUTE P.E.*S INTO ENERGY BOXES    ;;;;;;;;;
128 ---  C;;;; THEY ARE STORED IN ARRAY SPECT
129 ---     301 CALL BOXNUM(E1,E2,M1,M2,R1,R2)
130 ---  C????  FILL THE BOXES BETWEEN M1 AND M2 ?????
131 ---        IF(M2- M1) 42, 52, 53
132 ---     51 WRITE(6, 106) L, I, K
133 ---        GO TO 302
134 ---     52 SPECT(M1)= SPECT(M1)+ DSPECT
135 ---        GO TO 302
136 ---     53 DO60 M=M1,M2
137 ---        IF(M- M1) 42, 54, 55
138 ---     54 SPECT(M)= SPECT(M)+ DSPECT*(R1- E1)/Y
139 ---        GO TO 60
140 ---     55 IF(M- M2) 56, 57, 42
141 ---     56 SPECT(M)= SPECT(M)+ DSPECT*DEL(M)/Y
142 ---        GO TO 60
143 ---     57 SPECT(M)= SPECT(M)+ DSPECT*(E2- R2)/Y
144 ---     60 CONTINUE
145 ---    302 CONTINUE
146 ---  C*********   ION THRESHOLD LOOP ENDS HERE    ********** *****
147 ---    304 CONTINUE
148 ---  C*********   SPECIES THRESHOLD LOOP ENDS HERE   **********
149 ---      6 CONTINUE
150 ---  C---------   WAVELENGTH LOOP ENDS HERE   ----------
151 ---  C::::::::   STORE P.E. SPECTRUM IN AN ARRAY FOR THE 2-STREAM PROG. :::::
152 ---         TPROD=0.0
153 ---         DO 6086 M=1,100
154 ---         IM=101-M
155 ---         TPROD=TPROD+SPECT(M)
156 ---  6086   ELSPEC(J,IM)=SPECT(M)
157 ---         IF(SW1.EQ.0) GO TO 6088
158 ---        IF((IJ/10)*10.EQ.IJ) WRITE(6,100)
159 ---        WRITE(6,99) IJ,Z(IJ),TPROD,SZA(IJ)
160 ---  6088     CONTINUE
161 ---  C.... D+ PRODUCTION FOR 3 LOWEST STATES .....
162 ---         TPRION(1,IJ)=PRION(1,1)
163 ---         TPRION(2,IJ)=PRION(1,2)
164 ---         TPRION(3,IJ)=PRION(1,3)
```

```
      IF(ISW.EQ.0) GO TO 8
      WRITE(6,101) (CHI,(COLUMN(I),I=1,3))
      WRITE(6,105)(M,SPECT(M),M=1,125)
      WRITE(6,109)(PROD(I),I=1,3)
      PR4(1)=PRION(3,1)+PRION(3,4)+PRION(3,5)
      PR4(2)=PRION(3,2)+PRION(3,3)
      ENUM=0.0
      ENERGY=0.0
      DO 1493 M=1,125
      ENERGY=ENERGY+ENER(M)*SPECT(M)
 1493 ENUM=ENUM+SPECT(M)
      EBAR=ENERGY/ENUM
      WRITE(6,1463)
      WRITE(6,1464) (I,(PRION(I,K),K=1,7),I=1,3)
      WRITE(6,1464)(I,(PEHEAT(I,K),K=1,7),I=1,3)
      WRITE(6,1494) ENUM,ENERGY,EBAR,PICT
 8      CONTINUE
        IF(IJ.EQ.JMAX) GO TO 77
        IF(IJ.EQ.JPR) IJC=JMAX-2*J
   7  CONTINUE
C  %%%%%%%  ALTITUDE LOOP ENDS HERE  %%%%%%%%%
 77     JEMAX=101-JEM
        WRITE(8,330) JPR,JEM
        DO 78 L=JEMAX,100
 78     WRITE(8,678)  (ELSPEC(I,L),I=1,J)
 79     CONTINUE
   21   WRITE(6, 104)
        RETURN
 92   FORMAT(7F7.2)
 98   FORMAT(* ERROR 1,2 OR 3*,2X,I3,2X,I3,2X,I3)
  99  FORMAT(I6,F10.1,1P2E13.3)
 100  FORMAT(3X,*LEVEL*,5X,*HEIGHT KM.*,5X,*TPROD*,5X,*SZA*)
 101 FORMAT(10X,*CHI= *,F4.1,10X,*COLUMN DENSITIES*,//8X,*O2 *,1PE11.3,
     17X,*CO*,E11.3,7X,*N2*,E11.3 //)
 102  FORMAT((*0*, I3, 1PE12.4, 3(I6, E12.4)))
 103      FORMAT(*ENTER ELEC SPEC*)
 104  FORMAT(*0*, 10X, *NORMAL COMPLETION OF COMPUTATION* )
 105  FORMAT(35X,*PHOTOELECTRON ENERGY SPECTRUM*//(*0*,8(I4, 1PE11.3)))
 106  FORMAT(*0*, 15X, *L=*, I3, 5X, *I=*, I2, 5X, *K=*, I2, 5X, *DATA E
     1RROR*)
 108  FORMAT(1H1,*ELECTRON PRODUCTION RATES FOR:*/*    O2*,E15.7,/*     O
     1*, E15.7,/*     O *,E15.7,/*     HE *,E15.7)
 109      FORMAT(F9.2,8F7.2)
 110  FORMAT(126(1PE10.3))
 200  FORMAT(1F7.3,5I10)
 209  FORMAT(8F7.2)
 330  FORMAT(3I10)
 678  FORMAT(1P8E10.3)
1463  FORMAT(/10X,*EXCITED ION PRODUCTION RATES*/)
1464  FORMAT(10X,I4,5X,7E13.6)
1494  FORMAT(/10X,*ENUM=*,E12.4,2X,*ENERGY=*,E12.4,2X,
     1*EBAR=*,E12.4,*TOTAL HEAT INPUT=*,E12.4/)
 42   WRITE(6,98)I,K,L
      GO TO 302
      END
```

```
--- C.............<RSDG>.............
--- C+++++ THIS ROUTINE EVALUATES THE NEUTRAL COLUMN DENSITY
--- C++++ SEE SMITH & SMITH JGR 1972 P 3592, SUPR GTS3 IS THE MSIS
--- C++++ NEUTRAL ATMOSPHERE MODEL FOR Z<120 KM, CHI=SOLAR ZENITH
--- C++++ ANGLE, RE & GE RADIUS AND GRAV CON FOR EARTH
      DIMENSION XN(3),VERTCL(3),COLUMN(3),SN(5),M(3),DG(7),T(2)
      DATA A,B,C,D,F,G/1.0006963,0.5564383E-1,0.6819596E-1,0.7245E-3
     >                ,0.5649E-23,0.3E-51874/
        DATA    EM    ,   M(1) , M(2) , M(3)
     1    / 1.662E-24 ,    16  ,  32  ,  28/
      DATA  IDAY,   SEC  , GLAT, GLONG, GLST, F107A, F107, AP
     1    /0318, 51946  , 40.0 , 30.0 ,16.50, 75.0 , 71.0, 31/
C----IS SZA>90.0 DEGREES
      IF(CHI.LT.1.5708)GO  TO 2938
C----CALCULATE GRAZING INCIDENCE PARAMETERS
      ALTG=(6371.0E5+Z)*SIN(3.1416-CHI)-6371.0E5
C----IF GRAZING HEIGHT<120KM.,PRODUCTION IS ZERO --- MAKE COLUMN
C---- LARGE TO ACHIEVE THIS
      IF(ALTG.GT.120.E5)GO TO 1001
      DO 20 I=1,3
20    COLUMN(I)=1.E+60
      RETURN
C.... CALL NEUTRAL ATMOSPHERE FOR DENSITY AT GRAZING INCIDENCE
1001    ZG=ALTG*1.E-5
      CALL GTS3S(IDAY,SEC,ZG,GLAT,GLONG,GLST,F107A,F107,AP,48,DG,T)
      SN(1)=DG(2)
      SN(3)=DG(3)
      SN(2)=DG(4)
      TNJ=T(2)
C---- SN(1)=O , SN(2)=O2 , SN(3)=N2 , TNJ=TN
C---- GR=GRAVITY, RP=DISTANCE TO PT P, SH=SCALE HEIGHT,
C---- RG=DISTANCE TO PT G, HG=SCALE HEIGHT AT G
C
2938      CONTINUE
      GR=GE*(RE/(RE+Z))**2
      RP=RE+Z
        DO 10 I=1,3
      SH=(1.38E-16*TNJ)/(EM*M(I)*GR)
      XP=RP/SH
      Y=SQRT(0.5*XP)*ABS(COS(CHI))
      IF(Y.GT.100.0)WRITE(6,100)
  100 FORMAT(*WARNING, Y IN COLUMN(I) > 100*)
      IF(Y.GT.8) ERFY2=F/(G+Y)
      IF(Y.LT.8) ERFY2=(A+B*Y)/(C+D*Y+Y*Y)
    4 IF(CHI.GT.1.5708)GO  TO 2
      CHAPFN=SQRT(0.5*3.1416*XP)*ERFY2
      COLUMN(I)=XN(I)*SH*CHAPFN
        GO TO 10
    2 RG=RP*SIN(3.1416-CHI)
      HG=1.38E-16*TNJ/
     1    (EM*M(I)*980.*(6371.E5/(6371.E5+ALTG))**2)
      XG=RG/HG
      COLUMN(I)=SQRT(0.5*3.1416*XG)*HG*(2.0*SN(I)-XN(I)*ERFY2)
10      CONTINUE
```

```
    275 ---          RETURN
    276 ---          END
    277 --- C-------------------------------------------------------------
    278 ---                SUBROUTINE BOXNUM(E1, E2,M1,M2,R1,R2)
    279 --- C+++++ THIS SUBROUTINE FINDS THE BOX NUMBERS CORRESPONDING TO THE
    280 --- C     ENERGIES E1 AND E2, AND CALLS THEM M1 AND M2
    281 --- C+++++ THIS SUBROUTINE SETS UP ONE EV BOXES BETWEEN 1 AND 100EV
    282 ---       31 M1= E1+ 1.0
    283 ---          M2= E2+ 1.0
    284 ---          R1=M1
    285 ---          R2=(M2-1)
    286 ---          RETURN
    287 ---          END
    288 --- C-------------------------------------------------------------
    289 ---          SUBROUTINE PROBS(PROB,ZLAM,LMAX)
    290 --- C++++ THIS ROUTINE DETERMINES THE RELATIVE PROBABILITIES OF THE
    291 --- C+++++ DIFFERENT ION STATES -- SEE STOLARSKI & JOHNSON JATP 1972 P1691
    292 ---          DIMENSION NN(7),SUM(7),RAT(7),LAMD(7,4),SIGM(7,4),LAMM(7,4),
    293 ---         1  M(7,4),A(7,4),SIGIN(3,100),PROB(3,7,100),B(7,4),ZLAM(100)
    294 ---          REAL LAMD,LAMM,M,LAM
    295 ---          DO 9 L4=1,3
    296 ---          READ(5,100)  N
    297 ---          READ(5,100)  (NN(I), I=1,N)
    298 ---      100 FORMAT(7I5)
    299 ---          DO 10 I=1,N
    300 ---          K=NN(I)
    301 ---          DO 10 J=1,K
    302 ---          READ(5,101)  LAMD(I,J),A(I,J),B(I,J),M(I,J)
    303 ---       10 CONTINUE
    304 ---      101 FORMAT(F9.1,E12.3,2F9.2)
    305 ---          DO 20 LLL=1,LMAX
    306 ---          LAM=ZLAM(LLL)
    307 ---          TOTAL=0.
    308 ---          DO 19 J=1,N
    309 ---          SUM(J)=0.
    310 ---          JJ=NN(J)
    311 ---          DO 18 K=1,JJ
    312 ---          A1=LAM/LAMD(J,K)
    313 ---          IF (A1 .GT. 1.) GO TO 19
    314 ---       18 SUM(J)=SUM(J)+A(J,K)*((1.-A1)**M(J,K))*(EXP(LAM/B(J,K))-1.)
    315 ---       19 TOTAL=TOTAL+SUM(J)
    316 ---          SIGIN(L4,LLL)=TOTAL
    317 ---          DO 15 I=1,N
    318 ---          IF(SUM(I).NE.0) GO TO 16
    319 ---          RAT(I)=0.
    320 ---          GO TO 15
    321 ---       16 RAT(I)=SUM(I)/TOTAL
    322 ---       15 PROB(L4,I,LLL)=RAT(I)
    323 ---       20 CONTINUE
    324 ---        9 CONTINUE
    325 ---          RETURN
    326 ---          END

FINATION DATASET IS ABOUT TO BE WRITTEN
```

```
EDIT,D=PLIB,GN=R9PROD,DL
    1 ---    1.316          C          2         67         64         67
    2 ---   13.58   17.05   18.45   28.74   39.87
    3 ---   12.04   16.06   16.77   18.11   20.14   24.45   18.45
    4 ---   15.53   16.67   18.70   34.82
    5 ---  1025.70   12.10   12.10    3.50     .00    1.60     .00    1.00     .00
    6 ---  1010.20   12.20   12.20     .12     .00    1.40     .00    1.30     .00
    7 ---   991.50   12.50   12.50     .63     .00    1.90    1.40    1.60     .00
    8 ---  1008.50   12.10   12.50     .53     .00    2.80     .23    1.30     .00
    9 ---   977.00   12.70   12.70    4.66     .00   21.60   68.10    1.90     .00
   10 ---   972.50   12.70   12.70     .64     .00   21.90   68.10    1.90     .00
   11 ---   970.00   12.50   13.10     .28     .00   23.10   45.00    1.90     .00
   12 ---   949.70   13.10   13.10     .21     .00   30.10    7.16    2.30     .00
   13 ---   944.50   13.10   13.10     .09     .00   23.40   39.80    2.40     .00
   14 ---   937.80   13.20   13.20     .11     .00    3.20    3.15    2.60     .00
   15 ---   933.40   13.30   13.30     .10     .00   25.62    3.13    2.70     .00
   16 ---   930.70   13.30   13.30     .18     .00    3.54    3.13    2.75     .00
   17 ---   931.00   13.00   13.60     .08     .00   18.20   35.74    2.70     .00
   18 ---   901.00   13.60   13.90    3.04    2.96   12.50   46.00    3.40     .00
   19 ---   875.00   13.90   14.40    2.62    3.00   10.30   44.30    3.20     .00
   20 ---   850.00   14.40   14.80     .76    3.04   15.10   19.50    4.30     .00
   21 ---   833.50   14.80   14.80     .55    3.07   24.00     .00    6.50     .00
   22 ---   825.00   14.80   15.30     .75    3.10   28.70   13.70    7.70     .00
   23 ---   803.00   15.30   15.60     .20    3.10   40.70   21.30    9.50     .00
   24 ---   790.20   15.70   15.70     .32    3.11   31.20   29.20    9.60   17.70
   25 ---   787.70   15.70   15.70     .20    3.11   30.40   16.26    9.50    9.85
   26 ---   786.50   15.70   15.70     .09    3.11   30.20   16.26    9.60    9.84
   27 ---   780.30   15.90   15.90     .19    3.12   28.60   24.70   10.00   15.14
   28 ---   788.00   15.60   15.90     .16    3.11   31.10   37.10    9.60   19.70
   29 ---   770.40   16.10   16.10     .24    3.12   24.90    9.72    9.60    6.60
   30 ---   764.60   16.20   16.20     .14    3.13   22.70   35.50    8.50   23.40
   31 ---   760.40   16.20   16.20     .08    3.13   23.70   30.27    8.37   19.36
   32 ---   770.00   15.90   16.30     .15    3.12   24.40   26.20    9.20   18.20
   33 ---   755.00   16.30   16.80     .10    3.13   23.20   27.60    9.90   18.40
   34 ---   736.00   16.80   16.90     .03    3.13   31.60   29.60   19.00   25.20
   35 ---   703.40   17.60   17.60     .12    7.23   31.80   26.34   18.00   20.80
   36 ---   716.00   16.90   17.70     .13    7.10   32.90   33.60   18.60   25.30
   37 ---   665.70   18.00   18.00     .12    7.36   26.25   26.25   18.43   21.62
   38 ---   682.50   17.70   18.60     .24    7.35   26.20   31.20   22.30   25.60
   39 ---   647.50   18.60   19.70     .04    9.90   25.30   25.00   21.00   20.70
   40 ---   629.73   19.70   19.80    1.78   10.06   25.85   23.20   25.85   18.80
   41 ---   609.85   20.30   20.30     .55   10.16   26.13   23.10   26.13   18.64
   42 ---   615.00   19.70   20.70     .02   10.11   25.86   23.18   25.86   18.70
   43 ---   599.60   21.20   21.20     .16   10.20   26.62   23.11   26.62   23.11
   44 ---   584.33   21.20   21.20    1.31   10.24   22.01   23.11   22.01   23.11
   45 ---   554.51   22.30   22.30     .74   10.24   25.56   24.74   25.56   24.74
   46 ---   537.93   23.00   23.00     .12   10.20   25.20   25.30   25.20   25.30
   47 ---   507.90   24.40   24.40     .14   10.06   22.68   22.40   22.68   22.40
   48 ---   499.27   24.80   25.80     .00   10.00   22.05   21.98   22.05   21.98
   49 ---   540.00   24.80   25.80     .41   10.04   22.73   22.64   22.73   22.64
   50 ---   465.22   25.80   27.00     .23    9.71   20.39   21.84   20.39   21.84
   51 ---   476.00   25.80   27.00     .07    9.76   20.60   21.70   20.60   21.70
   52 ---   447.50   27.00   28.50     .08    9.52   20.00   22.00   20.00   22.00
   53 ---   417.50   28.50   31.00     .18    9.90   19.40   21.70   19.40   21.70
   54 ---   368.07   33.60   33.60     .64    9.26   18.35   17.95   18.35   17.95
```

```
  55 ---    360.76   33.50   34.90     .31    9.15   18.12   17.36   18.12   17.36
  56 ---    303.78   40.80   40.80    7.85    8.40   16.00   11.61   16.00   11.61
  57 ---    350.00   30.10   41.20     .42    8.72   17.38   15.00   17.38   15.00
  58 ---    284.15   40.00   44.30     .21    8.03   14.80   10.58   14.30   10.58
  59 ---    290.00   30.10   44.30     .37    8.22   15.22   10.80   15.22   10.80
  60 ---    274.24   45.20   45.20     .31    7.81   14.09   10.28   14.09   10.28
  61 ---    264.80   46.50   46.80     .24    7.59   13.44   10.01   13.44   10.01
  62 ---    270.00   44.30   47.70    1.61    7.77   13.95   10.11   13.95   10.11
  63 ---    256.37   48.20   48.20     .53    7.38   12.82    9.65   12.82    9.65
  64 ---    250.00   47.70   51.70    1.30    7.28   12.40    9.48   12.40    9.48
  65 ---    230.00   51.70   56.40     .43    6.42   10.70    8.35   10.70    8.35
  66 ---    212.50   56.40   60.50     .24    6.14    9.30    7.42    9.30    7.42
  67 ---    197.50   60.50   65.30     .71    5.71    8.35    6.30    8.35    6.30
  68 ---    185.00   65.30   68.90     .83    5.29    7.50    5.60    7.50    5.60
  69 ---    172.50   68.90   75.10    1.08    4.96    6.75    5.00    6.75    5.00
  70 ---    151.50   75.00   90.00     .22    4.30    5.32    3.80    5.32    3.80
  71 ---    120.50   90.00  120.00     .07    2.00    4.00     .90    4.00     .90
  72 ---      5
  73 ---      2       2       2       1       1
  74 ---    910.0      .752E-01    100.00      2.00
  75 ---    910.0      .452E 00     50.00     12.00
  76 ---    725.0      .841E-01    100.00      1.00
  77 ---    725.0      .540E 00     50.00      9.00
  78 ---    670.0      .720E-01    100.00      1.00
  79 ---    670.0      .309E 00     50.00      8.00
  80 ---    430.0      .229E-02     40.00      2.00
  81 ---    310.0      .407E-01     60.00      1.00
  82 ---      7
  83 ---      4       2       2       1       2       2       1
  84 ---   1026.7      .194E-01    121.70      1.00
  85 ---    900.0      .303E 00     44.00     10.00
  86 ---    850.0      .332E-19     13.70      4.00
  87 ---    750.0      .630E-26     10.00      3.00
  88 ---    769.8      .117E-15     16.40      2.00
  89 ---    769.8      .144E 00    144.80      1.00
  90 ---    736.8      .107E-15     16.80      1.00
  91 ---    736.8      .749E 00    166.80      1.00
  92 ---    682.3      .139E 00    126.90       .33
  93 ---    613.7      .201E-09     17.90      3.00
  94 ---    613.7      .814E 00    223.70      1.00
  95 ---    505.4      .226E-37      5.20      2.00
  96 ---    505.4      .412E-02     42.70      2.00
  97 ---    670.0      .365E-01     67.50      2.00
  98 ---      4
  99 ---      2       2       2       1
 100 ---    796.0      .356E-14     18.00      2.00
 101 ---    796.0      .652E 00     74.00      4.00
 102 ---    742.7      .391E-12     17.60      3.00
 103 ---    742.7      .329E 00     72.60      3.00
 104 ---    661.0      .986E-11     21.00      1.00
 105 ---    661.0      .172E 00    121.00      1.00
 106 ---    355.0      .151E-05     20.00      1.00
```

ITERATION DATASET IS ABOUT TO BE WRITTEN